



Designing Interactive Tools For Creators and Creative Work

Theophanis Tsandilas

► To cite this version:

Theophanis Tsandilas. Designing Interactive Tools For Creators and Creative Work. Human-Computer Interaction [cs.HC]. Université Paris-Saclay, 2020. tel-02941723

HAL Id: tel-02941723

<https://hal.science/tel-02941723>

Submitted on 17 Sep 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

DESIGNING INTERACTIVE TOOLS FOR CREATORS AND CREATIVE WORK

Habilitation à Diriger des Recherches

THEOPHANIS TSANDILAS

Chargé de Recherche @ Inria, Saclay
Laboratoire de Recherche en Informatique, Université Paris-Saclay

July 8, 2020

Jury Composition:

Sharon Oviatt	Professor, Monash University	Reviewer
Jürgen Steimle	Professor, Saarland University	Reviewer
Stéphane Conversy	Professor, ENAC, University of Toulouse	Reviewer
Géry Casiez	Professor, Université de Lille, IUF	Examinator
Fanny Chevalier	Assit. Professor, University of Toronto	Examinator
Wendy Mackay	Research Director, Inria & Université Paris-Saclay	Examinator
Patrick Bourdot	Research Director, CNRS, Université Paris-Saclay	Examinator

The Habilitation is a unique moment for the career of academics in France. Unfortunately, its use as a requirement for the direction of Ph.D. theses is an obstacle to the research potential of many young (and not only) faculty members. It further discourages some brilliant researchers from starting their career in France. This view is shared by many researchers with whom I had the chance to discuss in the past but is often forgotten when this turning point in a researcher's career is over. I hope that decision-makers will eventually understand the problem and decide to modernize the scope of an Habilitation thesis and its preparation process.

ABSTRACT

Creative work requires the learning and practice of advanced skills, perceptual, motor, cognitive, and aesthetic, but also relies on the use of diverse materials and sophisticated tools, both physical and digital. For these reasons, creative work is a major source of inspiration for research in Human-Computer Interaction (HCI). My habilitation thesis looks at the work of creators: artists with years of professional practice but also of learners, casual makers, and novice designers. I review the results of a series of studies, where I investigate how creators combine physical and digital representations and how these representations evolve throughout their creative process. My goal is to highlight key challenges for innovation in HCI and foresee how future interactive technologies can improve or extend current creation practices.

I focus on three widespread creation activities: drawing, physical modeling, and music composition. For these activities, I present examples of systems that assist the creation process in various ways: (1) they help novices practice well-established drawing-by-observation techniques through augmented photograph models; (2) they facilitate the image production of professional illustrators while they work on paper; (3) they enable modelers to synchronize their physical models with their digital representations; (4) they allow designers to embed interactive electronics in stretchable fabrication materials; and (5) they let composers of contemporary music extend their digital environments with personal vocabularies drawn on paper with physical ink.

The common goal of all these systems is to provide flexible representations, personal vocabularies, and effective interaction modalities that, on one hand, encourage exploration, and on the other hand, assist the transformation of early ideas to high-precision creative artifacts. Through the above examples, I discuss trade-offs between sketched-based representations that support expression and structured representations from which computers can easily extract meaning. Although the greatest part of the thesis concentrates on artistic activities, I also examine how informal, sketch-based representations can benefit other tasks, such as data-annotation and data-analysis tasks. I conclude with lessons for HCI research and reflect on directions for future work.

ACKNOWLEDGMENTS

I presented my Habilitation thesis remotely due to the COVID19 pandemic. Unfortunately, I did not have the opportunity to hug or shake the hand of my jury members and colleagues and thank them over a glass of red wine (preferably) or champagne. I will thus dedicate some space to express my gratitude here.

I will start by thanking my three reviewers, Sharon Oviatt, Jürgen Steimle, and Stéphane Conversy, first for their immediate responses and availability but also for their generosity, their thoughtful reviews, questions, and highly encouraging comments. I was extremely honored to have them as reviewers. I am also grateful to G ry Casiez and Fanny Chevalier. I have known them for more than ten years – Fanny is also a good friend. I have been following their work closely, admiring their brilliant careers. Many thanks to Patrick Bourdot who accepted to chair the meeting and deal with all the paperwork. I will not miss to give credit to Nicole Bidoit for her assistance throughout this long process. She was extremely responsive and present even during the confinement period. I would also like to thank St phanie Druetta for very similar reasons.

Special thanks go to Wendy Mackay, not simply because she was my “marraine” for this Habilitation thesis. I thank her for inviting me to be part of her team and for co-supervising with me two very successful Ph.D. theses. Wendy has been my mentor but also my longest-standing collaborator. Her influence is apparent in the work that I present in my manuscript. I will continue by thanking my colleagues and collaborators during these years, starting with Anastasia Bezerianos, who has been a close friend since we were Master students in Toronto. We prepared our Habilitation presentations together, and I was lucky to have her as my technical assistant during my presentation. Thanks to C dric Fleury and St phanie Huot, who have shared the same office with me for many years, Adrien Bousseau, an exemplary collaborator despite the distance, Pierre Dragicevic, for our passionate discussions usually about statistics, Themis Palpanas, an old friend as well with whom I found a way to collaborate closely despite our different research backgrounds, Michel Beaudouin-Lafon for his rare wisdom, but also my older colleagues at In-Situ Caroline Appert, Olivier Chapuis, and Emmanuel Pietriga, who made our lab a warm and enjoyable place.

I will not, of course, forget my Ph.D. students, J r mie Garcia, Michael Wesely, Anna Gogolou, and Arthur Fages, not only for their hard work but also for their wonderful characters. I thank the team’s wonderful assistants, Alexandra Merlin and Irina Lahaye, and the many people who passed from InSitu, ExSitu, AVIZ, and LRI during these years, as well as colleagues at Sophia Antipolis and Toulouse, where I spent a year. Those are only a few names of people who worked with me, shared their ideas, or helped me in different ways: Aur lien, Baptiste, Can, Carlos, Catherine, Christophe, Cl ment, Cl ment, David, Emmanuel, Emmanuelle, George, Halla, Ilaria, Joe, Karima, Krishnan, Lora, Magdalini, Mathieu, Ming, Olivier, Pierre, Rodrigo, Romain, Sarah, Thibault, Yvonne, and many others.

And to end, I will express my love and gratitude to my beloved Sandrine, a source of continuous inspiration, and of course to  lo se who has not stopped distracting me from my research.

CONTENTS

1	INTRODUCTION	1
1.1	Terminology and Scope	2
1.2	Overview	3
1.3	Methodology	5
1.4	Other Research Interests	6
2	DRAWING	8
2.1	Assisting Learners	8
2.2	Supporting Professionals	10
2.3	Conclusion	13
3	CRAFTING AND MODELING	15
3.1	Professional Modeling Practices	15
3.2	Novice Modeling Strategies	16
3.3	Assisting Modeling with Shape-Aware Material	19
3.4	Embedding Interaction in Stretchable Materials	23
3.5	Conclusion	25
4	COMPOSING MUSIC	27
4.1	Personal Grammars of Ink Gestures	28
4.2	Supporting Custom Composition Interfaces with Paper Substrates	30
4.3	Interactive Recognition of Handwritten Input	33
4.4	Observing the Use of Music Creation Tools	36
4.5	Conclusion	39
5	INTERACTION THROUGH SKETCHING	41
5.1	Drawing Interactive Traces on Paper	41
5.2	Sketching Widgets for Visual Exploration	44
5.3	Conclusion	48
6	CONCLUDING REMARKS AND FUTURE PERSPECTIVES	49
6.1	Drawing Assistance with Interactive Reference Models	50
6.2	AR-assisted Modeling	50
6.3	Sketch-Based Interaction Vocabularies	51
6.4	Designing with Data vs. Designing with Graphics	52
6.5	Methodological Questions	53
	BIBLIOGRAPHY	55

INTRODUCTION

Creative work has been at the core of research in Human-Computer Interaction (HCI). This is partly due to the great influence of Design methods in our field. Sketching, physical prototyping, and storyboarding are examples of interaction-design techniques that are directly inspired by professional practices in Arts and Design [Bux07]. Nevertheless, there are three other reasons why studying creative work is important:

1. Creators are keen users of technology. They are often demanding about their tools and are concerned with both artistic quality and performance. Hence, observing their practice reveals challenging problems that do not emerge when designing for the average user. Creators also engage in material practices and know the qualities and limits of traditional tools. Through long practice, they have mastered techniques to push their limits. HCI research has been increasingly interested in material practices [Ros+12] and their implications for interaction design [GK15]. By studying how artists work with physical materials, we can gain insight into how to transfer their use to the digital tools we design.
2. Creative work is not a privilege of professionals. Technology provides new opportunities for disseminating artistic practice to wider audiences. Video channels that teach drawing, crafting, and music creation techniques are followed by millions of amateurs or novices artists. We have also witnessed a rapid growth of the *Do It Yourself* movement [Tan+13], where digital fabrication tools and physical design remixing practices [OWM15] become more and more accessible to a larger public. Yet, acquiring skills to master an artistic activity requires significant investment in time. The challenge for HCI research is to integrate professional knowledge into the user interfaces that novices use.
3. Designing for an artist can help us generate design concepts that could serve other users. Framing a solution to a mathematical problem, planning an experimental procedure, or exploring a large dataset are all creative tasks. Supporting structure and precision can be crucial for such tasks, but it is equally important to provide cognitive aids that stimulate visual thinking and interactive tools that help users explore creative paths in their solutions. Past work has also shown that user interfaces that are closer to familiar interfaces, such as paper, can be more effective in supporting working memory and problem solving [OACo6]. In this thesis, I argue that interaction designers should target interfaces that are less rigid, more expressive, more familiar, and more personal.

I present the results of a series of studies that look at the work of creators. I am interested in artists with years of professional practice, as well as learners, casual makers, and novice designers. I study three different creation activities: drawing, physical modeling, and music composition. I focus on how creators interact with

and switch between physical and digital representations, and on how these representations evolve throughout their creative process. I present examples of systems that ease the transition from early exploration and free-form expression to more structured and more detailed representations. Through these examples, I investigate designs that extend the workspace of a computer with physical materials and tools. A key goal of the user interfaces that I discuss is to combine the practice of traditional techniques, e.g., physical modeling and paper drawing, with digital editing and end-user programming tools.

I also study tradeoffs between representations that support expression and structured representations from which the computer can extract meaning. I am motivated by how composers of contemporary music struggle between their personal music representations, as those are expressed on paper, and their formal implementations in their computer programs. I examine how sketch-based vocabularies can address such tradeoffs by allowing users to customize their interactions. Although I first focus on music composition, I then show how sketch-based vocabularies can benefit other tasks, such as data-annotation and data-analysis tasks.

1.1 TERMINOLOGY AND SCOPE

“Creativity” is possibly relevant to any human activity, from performing an instrument to analyzing some data and taking a decision. But my focus here is not creativity per se but the creation process, which includes the workflow, the techniques, and tools that creators use to produce a concrete artifact.

Next, I explain key terminology and further clarify the scope of the work presented in this habilitation thesis.

Materials (or Media). Some creators work with physical materials, e.g., paper, clay, foamcore, and wood; others use the computer or both. I focus on the intersection of physical materials and computer technologies. On one hand, I am interested in the editing capabilities and programming power of computers. On the other hand, my goal is to take advantage of the interaction qualities of physical materials, e.g., their support for direct physical manipulation and expression.

Tools & Devices. Traditional tools like pencils, brushes, and cutters are still widely used in today’s practice, often in parallel with machinery and electronic devices, such as printers, scanners, graphics tablets, milling machines, and laser cutters. I make a distinction between physical tools or devices, and software tools. Examples of software tools that I examine are drawing applications, 3D modeling tools, and music-programming software.

Representations. The term “representation” may refer to the medium that is used to depict an idea, concept, or artifact. However, it may also refer to the way that (“how”) the medium describes it. A representation can be physical, such as a physical model, or virtual, such as a 3D mesh. There is also distinction between representations that depict geometric relationships and aesthetics, such as drawings, and symbolic representations, such as musical scores. Symbolic representations may adhere to a specific structure (e.g., a graph or a tree) or a grammar (e.g., a musical score). Finally, a representation can be detailed and precise. Alternatively, it can be ill-defined or rough. I am interested in two different directions: (i) how to help creators move from rough

to high-fidelity representations, and (ii) how to combine multiple representations (e.g., virtual + physical or structured + sketchy) to support the creative task.

Creation Techniques & Creative Process. Any creation activity relies on years of accumulated experience about techniques and methods. In industrial domains, such as in automobile design [Bux+00], workflows are often standardized, following well-established uses of materials, tools, and representations. In other domains such as music composition, the work process can be very idiosyncratic. In both cases, the use of technology has an important role. Our challenge is not simply to design solutions that support current practices but also to further extend them with richer workflows that were not possible before.

Expertise & Skills. I study professional artists as well as learners, casual practitioners, and hobbyists. I will often refer to the first group of people as *experts*, and to the second group of people as *novices*. My goal is twofold: (i) to extend the expression vocabulary of experts, but also (ii) to create tools that transfer professional expertise to novices.

1.2 OVERVIEW

I split my study into four parts, where the first three are dedicated to creative work.

Drawing (Chapter 2). Drawing is perhaps the most common artistic activity that people start engaging in at a very young age. Still, realistic drawing is widely considered a privilege of a handful of skilled people. Despite this preconception, drawing is not simply a matter of natural talent; it relies on the use of well-established techniques developed through practice. The role of interactive technology is to disseminate these techniques and to further enhance them with computational support. I especially focus on drawing techniques that help first learners and then experts to progressively transition from rough sketches to detailed higher-quality drawings.

I first look at how drawing books and tutorials teach drawing-by-observation techniques. Such techniques rely on principles of scaffolding. The drawers start by approximating the main shapes of a drawing by sketching its overall structure. Then, they draw larger regions and continue with finer details, while frequently checking for alignments and proportions. I present an *automated drawing assistant* that assists learners in practicing this workflow as they draw from photograph models.

I then look at professional illustration and investigate how artists progressively move from early ideas to polished illustrations. I discuss why many illustrators still use physical media, e.g., paper, pencils, tracing devices, often in combination with computer drawing applications. I present *BricoSketch*, an augmented-reality system that helps artists progressively add details to their lower-fidelity drawings created with physical inks.

The first part of this research was conducted in collaboration with Adrien Bousseau (Inria, Sophia-Antipolis) and his Ph.D. student Emmanuel Iarussi. It was published at UIST 2013 [IBT13]. BricoSketch started as the Master

thesis of Magdalini Grammatikou, co-supervised by Stéphane Huot. We later extended this work, and our results appeared at ITS 2015 [TMH15].

Crafting & Physical Modeling (Chapter 3). Like drawing, modeling is a multiple-step process that requires the practice of well-established design techniques. The challenge now is to integrate the practice of such techniques into the digital tools that makers use. Specifically, I focus on how designers intermix sketches, digital models, and physical prototypes to create objects. I present a user study that simulates key phases of a structured design process (ideation, concept sketching, fabrication sketching, and physical prototyping) and observes how novice designers collaboratively transition from sketches to physical models.

A key outcome of this study is that material manipulation is important both for communicating problems and solutions, and for concretizing design ideas. Motivated by these results, I look at smart fabrication materials that could sense designers' manipulation of physical objects. I present two examples of such materials: (i) *ShapeMe*, a shape-aware material that senses its own geometry as it is cut by a designer; and (ii) *Stretchis*, stretchable silicone-based user interfaces that enhance physical objects with proximity sensing, touch sensing, and visual feedback. I discuss the implications of such interaction-aware materials for future CAD and fabrication tools.

The study on novice design practice was conducted in collaboration with Adrien Bousseau, Lora Oehlberg, and Wendy E. Mackay. It appeared at CHI 2016 [Bou+16]. *ShapeMe* and *Stretchis* were the results of Michael Wessely's Ph.D. thesis [Wes18], co-supervised by Wendy Mackay and funded by her CREATIV ERC grant. These results were published at UIST 2016 [WTM16] and UIST 2018 [WTM18].

Composing Music (Chapter 4). In contrast to drawing and crafting activities, music composition deals with non-visual, non-tangible concepts. The representations that composers use to describe music are often symbolic, but as other artistic representations, they also evolve as rough ideas become more concrete. I concentrate on composers of contemporary avant-garde music. These creators have a classical training but constantly try to innovate musical expression. As a result, they often develop very personal vocabularies to write their music.

I investigate how user interfaces that combine pen and paper can support such vocabularies. I present *Musink*, an ink-based language for augmenting paper scores with personal gestures that represent programmable musical objects. I also show how to extend music-programming tools with custom representations of musical objects printed or drawn on paper. Such representations are based on interactive, modular paper components that we call *paper substrates*. I discuss challenges concerning the automatic interpretation of handwritten content and present a mobile interface that helps musicians mitigate recognition problems. Finally, I discuss methodological challenges for studying music composition. I discuss the results of a user study that explores the use of pen-based technologies by 12 music composers.

Musink was the result of my postdoctoral research at Inria, in collaboration with Wendy Mackay, Catherine Letondal, and IRCAM. The work was

published at CHI 2009 [TLM09]. Jérémie Garcia’s Ph.D thesis [Gar14], co-supervised by Wendy Mackay, and Carlos Agon (IRCAM), extended this work. Results of Jérémie’s thesis that I will discuss here appeared at NIME 2011 [Gar+11], CHI 2012 [Gar+12], DIS 2014 [Gar+14b], and IHM 2014 [Gar+14a]. My work on the interactive recognition of hand-written scores was published at UIST 2012 [Tsa12].

Sketching Personal Widgets (Chapter 5). The need for custom representations is not unique to artistic creation. The fourth part of my analysis is dedicated to non-artistic tasks, focusing on sketch-based vocabularies. I present *knotty gestures*, a technique of pen micro-gestures that allow writers to attach active annotations to their notes on paper. I show how the subtle trace of such micro-gestures can be combined with other strokes (e.g., lines or characters) to add semantics and create links with the computer. I also present *SketchSliders*, range sliders that analysts can freely sketch on a mobile interface to customize their data explorations. *SketchSliders* take arbitrary shapes and support customization, annotation, and reuse. I investigate a range of specialized shapes, including circular sliders for periodic data, sliders with brunches that support multiple levels of granularity, and transformation sliders, whose shape controls the fisheye transformation of visualization views. I discuss how *SketchSliders* can support visual exploration in wall-display environments.

I worked with Wendy Mackay for *knotty gestures*, and our results appeared at AVI 2010 [TM10]. *SketchSliders* is the result of my collaboration with Anastasia Bezerianos and Thibaut Jacob. Our work was published at CHI 2015 [TBJ15].

I end my HDR thesis (Chapter 6) with a collection of related topics that I am currently working on and problems that I plan to study in the future.

1.3 METHODOLOGY

The research presented in this thesis is the result of teamwork and collaboration during a period of more than 10 years. It builds upon empirical results as well as the development of both software and hardware. This is a brief summary of our research methods:

Observation. We interviewed professional illustrators, architects, modelers, music composers, musical assistants, and biology researchers. Interviews usually took part in the work environment of our participants, where they presented examples from their artwork and demonstrated their work process. For a smaller number of participants, we made observations during longer periods (several months). We also tried to observe creative work in laboratory settings through well-structured but still creative tasks. Specifically, we designed a one-hour task to observe 12 music composers creating a musical piece. We also ran two one-day design charrettes to observe how novice designers sketch and prototype hand-fabricated objects.

Co-Design. Design workshops with individual creators or other professionals helped us further explore how our target users work and identify opportunities for

design. For our studies with composers, participatory design was an essential part of our design process.

Hardware & Software Engineering. As we often used technologies for which existing software solutions were limited or closed for industrial use, we had to develop our own software platforms. I will report on two software toolkits, which support the development of interactive applications. In addition to software, we developed custom hardware for sensing interaction with malleable fabrication materials, in particular with stretchable silicone and cuttable foamcore.

Evaluation & Experimentation. To evaluate our design solutions, we used a mix of methods, including user sessions with open-ended tasks and lab experiments. The former are better for observing divergent user strategies and assessing how users appropriate the tools. The latter are more appropriate for comparing concrete design solutions. In addition to user studies, we also conducted hardware evaluations to measure the accuracy and robustness of our sensing technologies.

1.4 OTHER RESEARCH INTERESTS

My HDR thesis describes on an important part of my research career but does not fully cover my past research activities and interests. I would like to highlight some significant work that I conducted in the last six years that is not discussed in the following chapters:

1. Our studies of perspective motor control [OTA14] with Halla Olafsdottir and Caroline Appert as part of our MDGEST JCJC ANR project (coordinated by Caroline). A substantial amount of research in Psychology has studied how people manipulate objects in the physical world, showing that they tend to use initial grasps that avoid uncomfortable end postures. We conducted three experiments to systematically observe how users plan both rotational and translational movements and analyzed our results in the light of the *Weighted Integration of Multiple Biases model* [Her13], a model of continuous heuristic planning. Our results provide insights about how to design techniques that facilitate grasping, but also techniques that can infer the user's planned movement and support it with early visual and motor aids (e.g., snapping). Our experiments focused on multitouch interaction on tabletops. Yet, I believe that our results have more important implications for complex manipulation tasks in VR and AR environments, where interaction is currently cumbersome and often problematic.
2. My work on statistical methods for measuring agreement in gesture elicitation studies [Tsa18]; [TD16], with significant help by Pierre Dragicevic. I consider it as my most mature and most significant contribution up to now. My TOCHI article has revealed serious flaws in the statistical methods introduced by well-cited CHI publications. It has further questioned the measures used by mainstream gesture elicitation methods to quantify consensus on gesture vocabularies (for a summary, see <https://agreement.lri.fr/>). As alternatives, the article proposes measures that are commonly used for the analysis of inter-rater reliability studies. This topic is still at the center of controversies, but I expect that they will be addressed by future publications.

3. My recent collaboration with Themis Palpanas and Anastasia Bezerianos for the Ph.D. thesis of Anna Gogolou (defended in November 2019), which concentrated on visualization and interactive similarity search techniques for large data series collections. During the first steps of Anna’s thesis, we experimentally studied how different visual encodings affect similarity perception in time series visualizations [Gog+18]. Our experiments provide empirical evidence that similarity perception is visualization dependent, where different encodings (color, position, or combinations of the two) are more or less sensitive to signal transformations, such as temporal and amplitude scale variations.

More recently, we have been looking at how to support data analysts in querying large-scale data series collections (hundreds of gigabytes in size), where answering a single similarity search query can take from a few seconds to dozens of minutes. Such delays are prohibitive for interactive visual-analysis scenarios. Although the database research community has introduced indexing techniques (see Themis’ recent survey [Pal20]) that provide quick approximate answers (in a range of a few milliseconds), such techniques do not provide any guarantee about the quality of their early approximate answers, i.e., how far from the exact one they are. Within the last year, we have developed statistical modeling techniques [Gog+19]; [Gog+20] that support probabilistic guarantees over approximate answers that progressively improve over time. As in other progressive analysis methods [FP16], our probabilistic guarantees can help analysts to decide whether to trust the current progressive answer, or wait for a better answer. We have further demonstrated that combined with a set of automatic stopping criteria, those techniques can lead to considerable time savings (e.g., 80 – 95% savings in time), while achieving high accuracy levels, e.g., 95% or 99% of their answers are exact.

DRAWING

Children start drawing from a very young age, but drawing is still considered a difficult task that requires advanced skills and years of practice. Drawing is not simply a recreational activity. It is essential for many professions in design, engineering, and visual arts. In this section, I investigate how interactive technology can support artists with years of professional drawing experience as well as casual drawers and learners. I will argue that drawing skills do not simply rely on talent. Through years of experience, artists learn how to practice well-established drawing techniques but also invent their own methods that help them enrich their artistic styles. We will see that drawing is rarely a single-stage process. Using rough sketches, hand-drawn guides, but also external materials, e.g., photographs, that act as models, is an essential part of any drawing activity. I will show how such techniques can serve as inspiration for designing novel interactive technologies that can assist the drawing practice of experts and learners. I will further examine the benefits of physical drawing materials and discuss how digital tools can benefit from them and integrate them into artists' workflows.

2.1 ASSISTING LEARNERS

A major challenge when learning *drawing from observation* is to trust what we *see* rather than what we *know*. Edwards [Edw79] argues that children confront an artistic crisis around ten as they realize that their abstraction of the world conflicts with their visual perception. Drawing books and online tutorials provide simple techniques to help learners understand how to draw from observation by gaining consciousness of the forms that they observe, their structure, and their relationships. Common techniques include drawing simple shapes as scaffolds – also known as *blocking in* – before drawing the subject of interest and checking for its alignments and proportions. Computers can disseminate such drawing techniques to larger audiences by providing interactive assistance and corrective feedback. Representative work in this area includes iCanDraw? [DPH10] and PortraitSketch [Xie+14] that help novices to draw faces, ShadowDraw [LZC11] that suggests completion of the drawing as it is performed, and Sketch-Sketch Revolution [FGF11] that automates the generation of step-by-step tutorials.

We started our research by looking at drawing books and online tutorials. Inspired by their recommendations, we set the following design goals [IBT13]:

1. Encourage learners to focus their attention on the actual model rather than their drawing.
2. Enable them to practice observation techniques proposed by the drawing literature. Such techniques can help novices identify shapes and understand their relationships and structure both in the original model and in their drawings.
3. Support corrective feedback to help learners anticipate their errors and refine their drawings.

We chose to focus on fundamental drawing techniques that apply to generic models, rather than domain-specific rules such as anatomy and perspective. We also focused on photograph models, as photographs are widely available and can be easily integrated into computer-assisted drawing systems. Photograph models can be further processed by vision-based algorithms and augmented with *visual guides* that help users construct their drawings.

To design our visual guides, we relied on three drawing principles that combine recommendations from books [Edw79]; [Dod85]; [HC11]; [Bra03] and tutorials [Hod12]; [Koh12]:

1. Lay down the main structure of the drawing with a coarse approximation of the main shapes.
2. Draw contours of large regions first and then details. The coarse structure created earlier serves as a scaffold that guides contour drawing.
3. Verify proportions and alignments to avoid or correct distortions.

We iterated on these principles with informal user tests on paper and the computer. Our final design supported the following forms of visual guides: (i) *block-in* lines, (ii) *skeletons*, (iii) *abstract regions* (or masses), (iv) line segments that show *alignments and proportions*, and (v) *grids*. Each visual guide provides a different level of assistance and may be more (or less) appropriate depending on the model. For example, block-in lines outline the outer bounds of contours. In contrast, skeletons emphasize the inner structure of a shape and are more appropriate for elongated structures and characters.

We used existing vision-based algorithms to automatically extract such guides from photograph models (see Figure 1). We then integrated these guides into a drawing user interface (see Figure 2). The interface is divided between two main areas: the model and the canvas area. The model area presents the photograph model to be drawn. The canvas area offers the drawing tools and a layered space for drawing the model. It further allows the user to choose among the available drawing guides, such as the block-in guides in Figure 2a.

A key feature of our system is its ability to evaluate error in the user's drawing. To evaluate error, it registers the visual guide shown on the model with the strokes drawn on the canvas. This registration is performed in real time. The user interface provides corrective feedback as the user draws on the canvas. In the example of Figure 2b, the system identifies a vertical misalignment, which is highlighted with a red dashed line. Our registration mechanism builds a dense correspondence between contours and a sparse correspondence between corners that we then use to support feedback about incorrect alignments and proportions.

As shown in Figure 2a, visual guides and corrective feedback are displayed over the original model. The user can then reproduce them on the canvas (by drawing) to use them as scaffolds. The rationale of this design was to encourage learners to actively practice the techniques. It further allows learners to refine the guides, instead of blindly following the automatically generated system suggestions. Our design encourages users to draw different guides in separate layers to facilitate registration and error detection.

We conducted two experiments to evaluate two aspects of our drawing assistant. The first experiment (12 participants) compared three versions of the drawing user interface on quick five-minute drawing tasks, where each version provided a different level of guidance: (I1) no guidance, (I2) guidance over the model, and (I3)

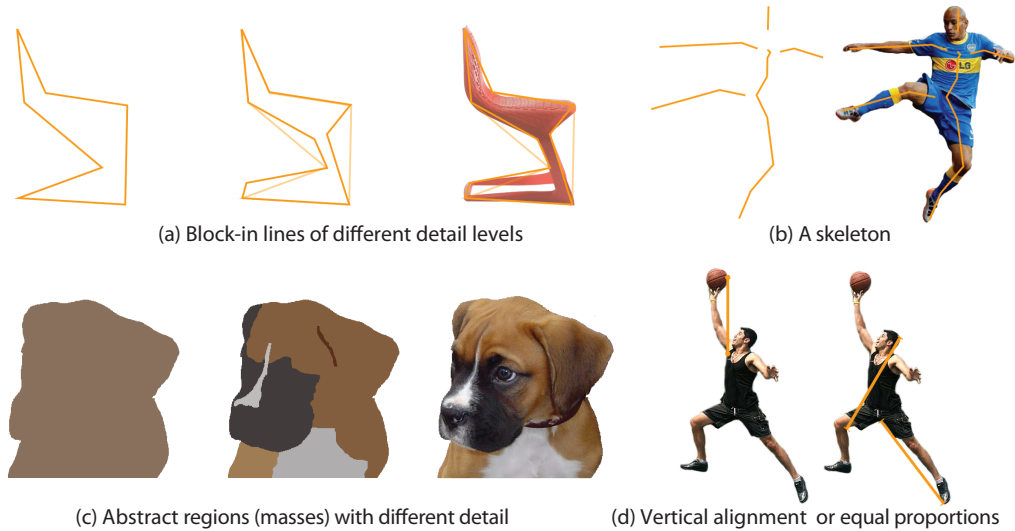


Figure 1: Drawing guides as they are automatically extracted from photograph models [IBT13].

guidance over both the model and the canvas. Full guidance (I₃) resulted in better drawings, reducing contour error by an average of 50% compared to the base user interface (I₁). Although guidance over the model (I₂) did not offer clear benefits in terms of precision, many participants appreciated the fact that it encouraged them to focus on the model and redraw the visual guides on the canvas by themselves.

A second experiment with eight participants examined longer drawing tasks (15 to 30 minutes), focusing on how over-the-model guidance can help users improve their drawings. Overall, guidance resulted in an average of 42%, 95% CI [19%, 65%] error reduction. Seven (out of eight) participants made extensive use of the visual guides. More interestingly, three participants who had experienced the guidance condition first tried to apply the techniques that they had practiced to the second no-guidance drawing task. For example, a participant tried to apply the signing and block-in techniques and reported:

“I could apply the methods on my second drawing, and I think they were very useful to better reproduce the photo. I understood clearly the interest of the explained method.”

Such feedback provides evidence that people can appreciate the visual-guiding techniques and quickly integrate them into their drawing practice.

2.2 SUPPORTING PROFESSIONALS

In contrast to novices, professional artists know how to draw by observation. They have further learned how to internalize geometric forms and relationships and generalize them when drawing from imagination. Even so, digital tools still offer opportunities for developing new methods of work. Not only do they allow artists to get inspiration from photographs, videos, or other artists, they also help them to accelerate their production through powerful editing and remixing tools. Nevertheless, the scope of computer tools is limited when artists use traditional media to complete their work. There are many reasons why some artists still use physical materials and tools, such as paper, pencils, and ink pens. Despite the progress

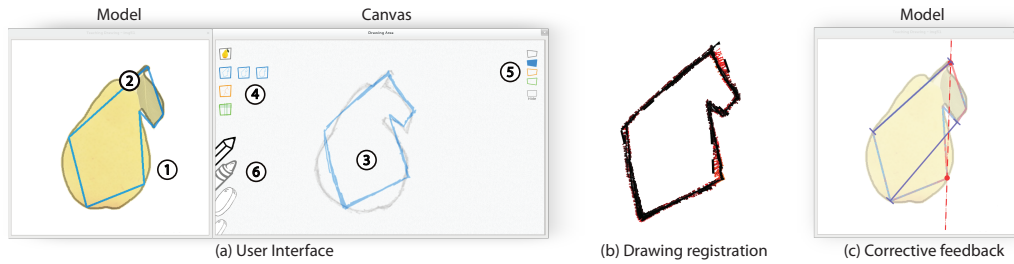


Figure 2: Overview of the user interface (a) of our drawing assistance [IBT13] and its mechanism of corrective feedback (b). The interface consists of a model area with the photograph and the visual guides and a canvas area. Here, the user has used a drop-down list (4) to activate a coarse block-in guide. The guide appears over the model in blue (2). The user has drawn the guide over the canvas in a blue layer (5) and then used it as a scaffold to reproduce a detailed contour (1,3). We offer basic drawing tools: a pencil, a pen, a small and big eraser (6). Our system registers the user’s drawing to estimate distortions (b) and shows erroneous alignments and proportions on the model (c). In this example, the red dashed line shows a vertical alignment that has not been respected by the user. The dark blue segments highlight two distances that should be made equal.

of artistic stroke-rendering techniques [Her03], current computer tools cannot capture the richness and variety of artistic styles supported by physical media [NSI14]. Therefore, artists still rely on paper for styles and techniques that software tools do not currently offer.

Our work for *BricoSketch* [TMH15] allowed us to study how professionals combine interactive technologies with traditional drawing tools and explore how computerized tools can provide assistance while working on paper. We started by conducting interviews with four illustrators in Paris. Our participants were freelancers with 4 to 11 years experience as professional illustrators. Their projects included illustrations for books, magazines, and newspapers. Three of them also worked as writers of graphic novels. We identified a variety of different ways in which artists use traditional and digital drawing tools in their projects. Sometimes, almost all the work is done on paper. For other projects, paper is never used. In other scenarios, artists switch from paper to software but also from software to paper in different variations. Which strategy to choose depends on the artistic style that the artist wants to achieve, the quality of the expected result, as well as the drawing speed that each approach affords. Speed is often a key constraint as artists are often asked to complete a task within limited time. For example, it is not uncommon for newspapers to ask for one or several illustrations within the same or the following day.

We observed that projects evolve at multiple stages where early sketches and drawings often serve as templates for higher-fidelity ones. Artists use either physical layers on top of a *light table* or virtual ones in computer software to make copies of their images and draw new versions on top of them. Figure 3 presents examples of techniques used by our study participants. In Figure 3a, the artist uses a home-made light table to draw the detailed version of a graphic novel. An earlier rough version of the page serves as a template in this case. Figure 3b presents a different approach for making the transition between early sketches and final pictures. This artist creates sketches through dark "masses" created with Adobe Photoshop. These sketches are printed in light blue, helping the artist to draw the detailed

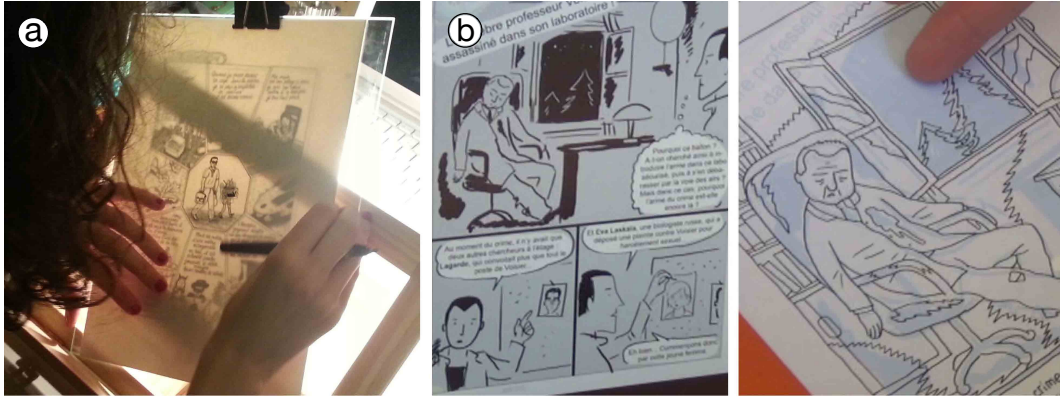


Figure 3: Techniques used by illustrators to shift from sketches to high-quality pictures: (a) Drawing with a home-made light table. The artist draws the panels of a page for a graphic novel by using earlier sketches as guides. (b) A different artists creates sketches on the computer as dark shadows (left). The artist prints them in light blue and uses the same page to draw the detailed contours with a pen ink (right). The light blue shadows can be easily removed when scanning the page.

contours with a black pen ink. When scanning the page, the light blue color is removed, and the artist can add colors over the detailed contours. In both cases, the artists' approach is driven by their need to draw the contours with a physical pen, as according to them, its trace has clear quality differences compared to the digital ink that can be achieved with a graphics tablet. Both these artists intermixed both physical and computer drawing tools for different stages of a project (i.e., early sketching, drawing detailed contours, coloring), depending on the requirements of each project.

We also observed the work of an illustrator for a graphic novel for several months. For this project, the artist had to draw all images with a pencil. Unfortunately, producing high-quality images with this style was very time consuming, and the artist experimented with various solutions to deal with time constraints. She eventually developed a method of combining and reusing drawings of different resolutions and scales to create her final pages (see Figure 4a). Although sufficiently quick for the time and style constraints of this project, this process required significant manual effort. The artist had to scan and print her drawings several times in order to switch between paper and digital versions of her work.

In order to assist such workflows, we developed *BricoSketch* [TMH15], an augmented-reality paper-based system. *BricoSketch*'s concept (*bricolage* + *sketch*) is based on the observation that professional illustration does not solely rely on sketching and drawing skills. Artists often have to engineer scaffolding and remixing techniques to increase their productivity or produce new artistic styles. Figure 4b-d shows a scenario of use of the system that captures the workflow discussed earlier. The artist's workspace contains the main illustration, drawn on paper with a pencil, and several rectangular areas that represent partial virtual views of the original drawing displayed with an overhead projector. The user creates and transposes the views interactively by moving the pencil over the drawing surface. In this example, the artist has created "transposed" views to decorate the labels of ingredient containers. The scale of these views is larger so that the artist can draw small parts of the image with higher control and finer detail.

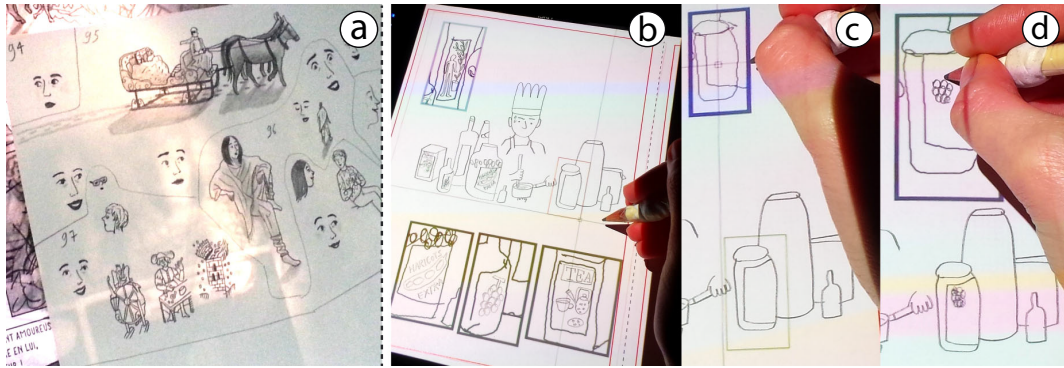


Figure 4: (a) The artist uses a light table to draw higher resolution features, such as important details and facial expressions. The final pages are constructed by digitally merging all the pieces together. (b-d) *BricoSketch*: The artist uses the system to add details to different parts of a composition that shows a cook preparing a meal. She uses a pencil to select a jar in her drawing (b), create a larger copy (c), and then draw higher resolution details (d).

As shown in the figure, the physical ink of the pencil within a view is blended with projected copies of ink drawn in its other views. In order to produce the final digital illustration, the artist needs to blend the partial images of the physical ink. Blending can be performed manually by using common image-editing software. Alternatively, we can partially automate it by inferring the correspondence between the virtual workspace and the scanned illustrations and then letting the user decide about how to blend all pieces together.

The implementation of *BricoSketch* was based on *iSketchnote*¹, a pen tracking technology that captures the 3D position and 3D orientation of small magnetic rings. A ring can be fixed on a pen or a pencil to detect the position of its tip as a user draws on paper. This technology had limitations, in particular it suffered from spatial deformations that could not be fixed. However, it was the only one to support custom drawing tools at the time. Furthermore, since the technology can sense the position of the pen tip above the drawing surface, it allowed us to support pen-based interactions that do not leave a trace on paper, e.g., to create, position, and scale the virtual views of an illustration.

We asked an artist to use *BricoSketch* in two consecutive sessions. In the first session (40 minutes), we introduced the system, refined our command shortcuts, and chose a pen and pencil. In the second session (1 hour), the artist used the system to produce two illustrations. The artist found that the system is an excellent tool for exploring ideas and trying out alternative versions of drawings. She explained that it helps to quickly test ideas by reusing parts of her drawings directly on paper. She also proposed that copies can serve as templates for drawing new variations. For example, in a sequence of panels for a comic strip, the artist could copy the face of a character and use it as model to draw its variations in other panels.

2.3 CONCLUSION

Drawing assistance is equally important for learners who want to improve their drawing skills, and for professionals who seek ways to speed up performance but preserve their unique artistic styles. Drawing is often a multi-step process: the artist

¹ <http://www.isketchnote.com>

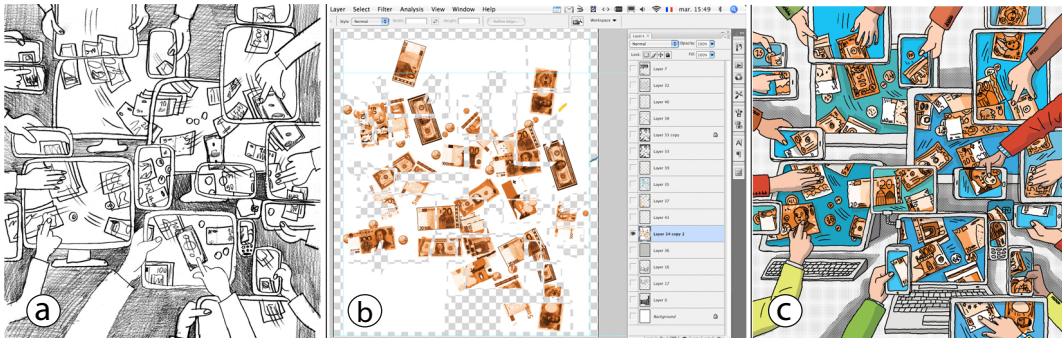


Figure 5: Use of images as models for more realistic drawing: (a) Initial sketch created with pencil on paper. (b) The artist's workspace in Photoshop containing several overlapping layers. The active layer shows cropped images of banknotes that serve as models for drawing the final illustration on the computer. (c) Final illustration realized with a graphics tablet. Artwork by ©Sandrine Martin.

starts with rough, ill-defined representations and progressively moves to more detailed drawings. Interactive technology can assist this process in many ways: (i) reinforce the practice of well-established scaffolding techniques, (ii) provide corrective feedback, (iii) support reuse, (iv) facilitate the mix of different representations (either digital or physical), and (v) provide models in context for reference. We learnt from our interviews that artists often make use of photographs, films or even sceneries and physical objects not only for ideas and inspiration, but also as direct support for their drawings. Famous illustrators like Norman Rockwell are well known to make extensive use of model photographs in their work. Figure 5 presents an example from the workspace of an artist working on a newspaper illustration. The artist has created multiple layers on Photoshop to integrate her early sketches that provide the overall structure of the drawing but also images (here bank notes) collected on the internet to be used as reference models. Another interesting practice that we observed is the capture of self photos to create models of complex body postures or grasps. How to assist such practices is a future direction that I discuss in Chapter 6.

In the following chapter, I investigate physical modeling tasks that often involve 3D shapes. Again, I am interested in how creators combine modeling representations to perform such tasks, focusing on the use of physical materials and tools.

CRAFTING AND MODELING

Like drawing, crafting requires artistic skills and also aims at creating shape. Crafting is a common activity of modeling practices in industrial design and personal fabrication. Even though modern 3D software allows artists to create high-quality 3D models by fully working on the computer, well-established modeling practices often rely on physical representations, including drawings on paper and physical prototypes.

In this chapter, I focus on physical modeling methods, but I am particularly interested in the transition between physical prototypes and their alternative representations, either sketches or computerized 3D models. I start by discussing how professionals use physical models in their design practice. I continue with the results of an observation study that looks into how novice makers transition from early sketches to physical materials and hand-fabricated objects. The following sections focus on a technical challenge, which is how to make fabrication materials interactive. I present two material-driven technologies that address two different aspects of this problem: (1) how to fabricate physical models that can sense their own shape; and (2) how to embed interaction in highly-stretchable materials.

3.1 PROFESSIONAL MODELING PRACTICES

Physical modeling is widely practiced in professional design despite the parallel use of powerful 3D modeling tools, such as AutoCad, Maya, Rhinoceros 3D, or Blender. Professional design projects often group together experts with different but complementary skills. In this case, a design process may require them to iterate over sketches that capture different design aspects, physical prototypes of various scales, as well as digital models. For example, in interior and exterior automobile design¹, the use of physical (e.g., clay) models of various scales is a well-established practice. Such models help design teams to iterate on their designs by moving from sketches and design concepts to detailed prototypes. The methods that automotive modelers use to “sculpt” their 3D clay models has inspired interaction techniques such as 2D and 3D digital tape drawing [Bal+99]; [Gro+02]. Despite the introduction of novel interaction displays, workflows that involve physical models still persist today.

We have conducted interviews [WTM18] with other practitioners who use digital models and physical prototypes: architects, modelers, and an industrial shoemaker. In architectural design, physical models (maquettes) were primarily used in the past for communication with clients and for demonstration purposes in competitions. Their role has now changed, since other media such as videos are more effective (or more appealing) in demonstrating architectural projects. Even though, physical models are widely used within design teams to iterate on ideas and concretize designs. The emergence of affordable digital fabrication machines and fab labs has recently accelerated their construction. A senior modeler of a large archi-

¹ We had the chance (with Adrien Bousseau and Wendy Mackay) to interview and observe automobile designers on early concept creation. Results from this study were confidential and have not been published.

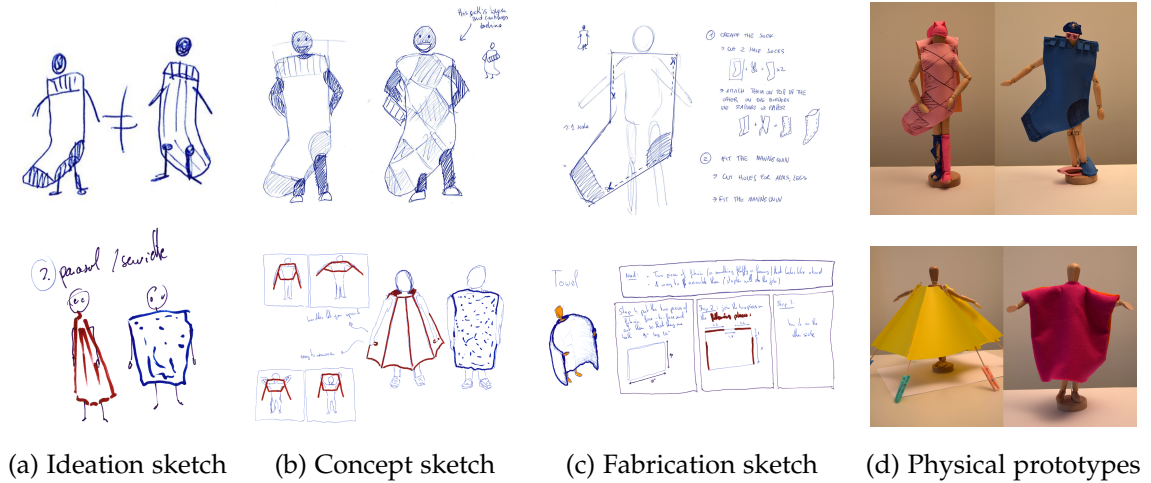


Figure 6: We ran two all-day design charrettes [Bou+16], where participants sketched and collaborated to prototype pairs of costumes. The figure presents representative sketches and prototypes for the different phases of the study.

textural firm told us in an interview that the use of laser cutters and 3D printers have allowed them to “stop working on weekends.”

Although digital fabrication machines have helped designers to produce high quality physical models quicker, the digital fabrication workflow is still far from being interactive. According to Baudisch and Mueller [BM17], the current use of fabrication machines resembles to computer use in the 50s, when input and output was largely performed with punched cards. Part of my focus in this chapter is how to make the fabrication process more flexible and more interactive.

3.2 NOVICE MODELING STRATEGIES

Fab labs have made fabrication available to a new population of casual non-professional makers who can now participate in the creation and customization of wide variety of physical objects [RB09]; [Mot11]; [Tan+13]. Neil Gershenfeld argues [GGCG17] that custom fabrication will continue to explode with an exponential growth that is similar to Moore’s law. Unfortunately, most CAD tools have been developed based on professional design practice. Even though some enthusiastic makers take the time to learn professional design software, novices may find them too complex and effort demanding. An additional difficulty that novices often face is the lack of training and intuition about how to develop design concepts, iterate on them, and translate them into detailed models. Some casual makers are keen on solving construction problems but have poor design skills. Others are good at drawing down ideas but have little intuition about how to turn their sketches into 3D models, how to choose materials, or how to deal with engineering problems.

Our goal was to understand how to help people with no formal training to make objects, focusing on the transition from sketching and designing ideas on paper to fabricating physical prototypes. We also wanted to study how such *novices* collaborate with each other and how they use sketches, fabrication materials, and prototypes to iterate on their designs. Specifically, we were interested in the following questions:

1. **How do novices sketch for themselves and for others?** The role of sketching in creative thinking has been studied in a wide range of disciplines, including design research, cognitive science, and cognitive psychology [PG98]. Eckert et al. [Eck+12] identify three main roles of sketches: to generate and record ideas, to represent abstract properties pictorially, and to communicate design ideas to others. They observe that not all designers sketch to generate ideas, but for many designers, a fundamental role of sketches is to quickly communicate with others. Others have looked at differences between novices and experts. Suwa and Tversky [ST97] examine how professional architects and students think over their sketches and find that experienced architects could identify more functional relations in their sketches and pursue with them deeper design thoughts than students. Ahmed et al. [AWBo3] observe that beginners may lack confidence in their decisions and tend to adopt “trial and error” strategies. Beginners also favor 3D visualization or physical manipulation of models to understand their function and assembly. In contrast, experts tend to evaluate intermediate solutions to decide if they deserve further implementation. Such evaluations allow experts to identify potential issues but also to refer to past designs that have addressed similar problems. Finally, Cross [Cro04] mentions that when problems are well-defined (e.g., when solving routine problems or playing chess), novices often adopt “depth-first” strategies, while “the strategies of experts are usually regarded as being predominantly top-down and breadth-first,” which leads to easier solutions. However, in creative domains like design, Cross argues that “creative experts treat problems as ‘harder’ problems than novices do.”
2. **How do novices fabricate prototypes with hands-on materials?** As I discussed in the previous section, designers often use physical models to represent or explore design ideas. According to Faas et al. [FBY14], quick physical prototypes are as effective as quick sketches in shaping designs. Prototypes also serve as a means to iterate on a design. For example, Dow et al. [DHK09] find that iterative prototyping produces higher-performing design concepts than non-iterative prototyping. Brereton and McGarry [BM00] observe engineering students and argue that physical objects and prototyping materials themselves can support design thinking by serving: (i) as starting points, (ii) as thinking props that afford specialized actions, e.g., grasps or other gestures, (iii) as memory devices, (iv) as physical embodiments of abstract concepts, (v) as interaction probs for exploring parameters, (vi) as supports for understanding relationships among quantities and revealing practical limits, (vii) as within-group communication artifacts, etc. Wendrich [Wen10] reports on observations of design students using various CAD, graphical sketching, and physical prototyping techniques. He reports that tangible interactions with physical construction material adds quality and detail to the end result and enhance participants’ concentration and involvement.
3. **How do novices collaborate for fabrication?** Collaborative design may require constant communication through sketches and prototypes. Dow et al. [Dow+11] show that when designers share multiple prototypes with peers, they achieve better rapport in the group and better quality results. But in addition to visual and physical artifacts, collaborators also communicate via hand gestures, gaze, body gestures, and visual cues [SM14]. Such communication modes become important when supporting teams collaborating at a distance. For ex-

ample, in a distributed construction task, Kirk and Fraser [KSF06] find that gesturing using hands is quicker than using digital sketches without any loss of accuracy. Eris et al. [EMBS14] examine the role of gesturing during design sketching and observe that sketching and gesturing may play different roles in different phases of a compressed design process. Gesturing occurs earlier when exploring the problem while sketching emerges later when detailing the identified concepts.

In order to answer these questions, we ran a study [Bou+16] that consisted of two all-day *design charrettes*, in which 12 participants (six participants per charrette) designed and hand-fabricated pairs of objects (see examples in Figure 6). We asked the participants to develop mock-ups of costumes on doll-sized mannequins. We provided a range of prototyping materials and tools as well as small mannequins on which they could build their costume models. For sketching, we provided blank A3 pages and a variety of pens, colored markers, pencils and erasers.

We designed the charrettes to have both an open-ended creative task and well-defined stages and outcomes, either sketches on paper or physical prototypes. This approach let us observe pairs of participants as they successively worked individually and together through several phases of the design process. Furthermore, it allowed us to structure and control key aspects of the design process, e.g., design phases, materials and tools, collaboration roles, discussions, while collecting rich data from a range of sources such as sketches on paper, physical models, and videos of collaboration discussions.

We structured each design charrette as four main subtasks that emulate the common phases of professional design process: (1) ideation sketching, (2) conceptual sketching for presentation, (3) sketching for fabrication, and (4) fabrication. Participants worked individually for all but the fourth phase. For this phase (fabrication), the participants formed pairs, and each took the role of a leader or an assistant. The goal of the leader was to build one of the two paired objects (costumes) that he or she designed. The goal of the assistant was to build the other object, following the instructions of the leader. We controlled their collaboration by means of structured discussion sessions that interleaved with sessions of individual work time, when participants could make progress on their prototype. Each one-hour fabrication round consisted of three five minute discussions followed by three 15-minute individual work sessions. Our coding scheme captured both textual elements (e.g., titles, instructions, annotations) and graphical elements in the sketches. Before data analysis, we coded all sketches produced by our participants. We also coded discussion videos and recruited 10 external evaluators to evaluate the final physical prototypes while also inspecting their fabrication sketches.

We discuss here some key findings and implications of the study. Our participants used diverse drawing techniques (multiple views, perspective, unfolding and exploded views) to describe their concepts across design stages, even though they did not necessarily have the ability to draw them accurately. Automated drawing guidance, as in the approaches that we discussed in the previous section, could possibly compensate for the limited drawing skills of casual makers and help them adopt techniques of professional designers. Our analysis of sketches and prototypes suggests that higher quality instructions (as integrated into fabrication sketches) lead to prototypes that better reflect their designs. It also seems that detailed sketches capture concepts that are well finalized, and as such they can be easier to reproduce as a physical model. In contrast, concepts described by coarse

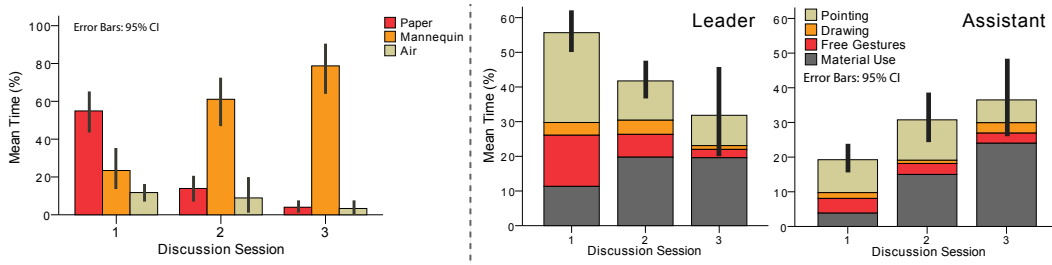


Figure 7: Summary results on the use of paper, physical materials, and gestures during discussion sessions [Bou+16]. Left: Use of sketches on paper, the mannequin with the fabrication model, and the air as reference for each of the three discussion sessions. We measure time as percentage of the total time of discussion. Right: Use of different types of actions by the leader and the assistant. Leader actions decrease over time, while assistant actions increase. Material use progressively replaces pointing and free-hand gestures.

sketches need to be refined during prototyping. Our analysis of discussion sessions indicates that participants tried to overcome the lack of information in fabrication sketches by spending more time discussing details over the physical models.

Figure 7 provides additional details about our analysis of the three discussion sessions. The results show that the reference of discussions progressively moved from paper to the mannequin, while prototyping material eventually became the dominant medium of collaborators’ actions. Mid-air gestures also reduced as participants started using the prototypes as reference during discussion. We further observed that fabrication materials often guided the fabrication sketches. According to a participant:

“I had ideas about fabrication once I saw the material. I also got inspiration by seeing how others did.”

Moreover, the mannequins served as models that allowed participants to explore poses or verify dimensions and proportions.

Overall, a key finding of this study was the critical role of physical materials in the transition from sketch-based models to prototypes. Materials supported design exploration and testing during the sketching phase and even supplanted sketches during discussion on physical prototypes. Some participants did not anticipate the fabrication challenges while they sketched their costume designs and only resolved them through physical manipulation of fabrication materials. This observation motivates the need for “tangible” CAD interfaces that enable designers to model 3D objects by interacting with physical prompts. Since novices lack good intuitions about how to hand-fabricate 3D shapes from raw materials, it is also important to develop “material-aware” solutions that sense material manipulations (e.g., cutting and folding paper and foamcore) and assist makers when they practice crafting techniques. I investigate this direction in the following two sections.

3.3 ASSISTING MODELING WITH SHAPE-AWARE MATERIAL

Back in 2011, Willis et al. [Wil+11] described *interactive fabrication* as the digital fabrication approach where makers interact with a physical model while their crafting input is captured in real time by the computerized system. Past HCI research has tried to support interactive fabrication with a range of different systems:

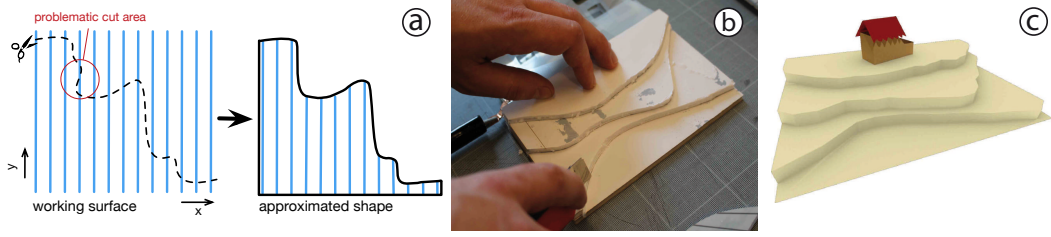


Figure 8: Overview of the ShapeMe [WTM18] technology. We approximate 2D shape by using a grid of length-aware sensors. The red circle highlights the part of the cutting path that is not accurately captured by this sensing topology (a). The approach allows makers to craft physical models that consists of multiple shape-aware layers (b). The geometry of the model is constantly captured and rendered in 3D modeling software, such as Blender (c).

Vision-based systems that use 3D scanners [Fol+10]; [Wei+15a] or depth cameras [Piy+16] to capture shape.

Modified fabrication machines such as Constructables [MLB12] that support direct user input or augmented-reality fabrication systems [Pen+18].

Smart crafting tools, including hand-held milling devices [ZP13], augmented 3D extruder pens [Yue+17], and Anoto pens [Son+06].

Modeling proxies that can detect their deformations or their 3D topologies [LRL17].

Each approach has its own strengths and limitations. Vision-based approaches require a setup of external cameras and suffer from occlusion problems. Smart tools assume that the physical model remains fixed, or that additional calibration mechanisms are required. Other solutions pose constraints on the construction materials that the designer can use. Our goal was to make materials aware of their own shape rather than delegating sensing to external devices or systems. We refer to such material as *shape aware*. ShapeMe [WTM18] is a sensing technology that achieves this goal, focusing on cuttable fabrications materials e.g., paper, soft wood, and foamcore.

To approximate shape, ShapeMe relies on grids of printable sensors that take the form of thin lines (or polylines). Sensors are positioned in a two-dimensional space and can sense their length. If we also know the position of the sensors in space, we can approximate a 2D shape. In the example of Figure 8a, sensors are placed in parallel along the x axis, and the surface is cut by following a curve. The sensors' positions provide the x coordinates of the curve's points, while their lengths provide their y coordinates. This principle can be extended to 3D objects by stacking several layers on top of each other (see Figure 8b). Consider that creating volumetric objects out of thinner sheets is a well-established model-making practice [Wer11]. A major strength of our approach is that information about the final shape of a crafted piece is persistent, even when sensors are disconnected or offline (its history may be lost though). An additional strength is its robustness to occlusions problems.

We implemented length sensors as custom-made capacitors. To produce a capacitor, we print two layers of conductive material (silver nanoparticle ink and PEDOT:PSS) in close distance, separated by an insulating layer that serves as the dielectric. The length of a line-shaped capacitor is a linear function of its capacitance

and can be indirectly estimated through a voltage-divider circuit. By appropriately choosing the circuit elements, we achieve a close-to-perfect linear relationship between the length of sensors and voltage. This linear relationship allows us to estimate the length of a sensor with an error of $\pm 2 - 3$ mm (for 1mm-wide sensors). Part of this error is due to imperfections of our fabrication process. A major constraint in our implementation was the lack of double-coated sheets that could be inkjet-printed onto both sides. We thus relied on screen-printing (see Section 3.4) for the back side of the sensors, which introduced additional sources of error. However, we believe that precision can be greatly improved in future iterations, when high-quality printing on double-coated sheets becomes available.

To connect and control grids of multiple sensors, we designed a custom printed circuit board (PCB) that multiplexes analog input from up to 64 sensors. Multiple *ShapeMe* boards can be stacked together to multiply the number of sensors. Our implementation enables a standard Arduino Uno controller to process up to 384 sensors, while more advanced controllers can theoretically support a larger number of sensors. The number of sensors on a single sheet is of course constrained by their width (> 0.25 mm) and their distance (> 1.5 mm).

A main drawback of *ShapeMe* is that the sensing precision is further constrained by the shape and orientation of the sensors. As shown in Figure 8a, if a sensor is cut at multiple positions, then sensing can fail. We explored a number of solutions to this problem, including sensors that take alternative shapes, e.g., curved lines, and branched sensor structures. Unfortunately, early experiments showed that capacitance is greatly affected by the shape and curviness of the sensors. We thus decided to focus on sensing topologies with sensors consisting of one to three straight-line segments, as such sensors result in more reliable length estimates.

Our approach assumes that even when makers do not a-priori know their target shape, they have a rough idea about the parts of the model they need to work on and the direction of their cuts. Thus, they can pre-configure the topology of their shape-aware material according their needs. Figure 9 shows examples of three sensor topologies that we currently support. The first topology consists of vertical sensors and is optimal for horizontally oriented cuts. The second has a star layout and is optimal for sensing cuts around the periphery of the object. Finally, the third is best for sensing vertical cuts but also allows to detect holes. How to create optimal sensor topologies for sensing cuts is a problem that we have not fully addressed. Future work could make several contributions in this direction, e.g., by exploring more sophisticated empirical models that deal with curviness, providing computational methods to optimize the layout of sensors, or producing hardware that combine multiple layers of *ShapeMe* sensors with different layouts.

Designing *ShapeMe* sensors, fabricating them, and deploying their applications would require considerable time and effort if it was not assisted by a dedicated design tool. Figure 10 presents the user interface of our software toolkit as it communicates with the Unity platform. The toolkit helps makers to (i) digitally create an initial geometry, (ii) design a sensing and a wiring structure, (iii) export a *ShapeMe* model to be printed with a laser cutter and print its sensors, and (iv) synchronize a physical model with external 3D modeling software. As shown in the figure, a model consists of one or multiple layers formatted as printable A4 or A3 sheets. Each layer represents the 2D layer of a 3D structure, or alternatively, an individual 2D surface, such as a wall or the ground of a house. Our software communicates with external digital modeling through the Open Sound Control (OSC) protocol [Wrio5]. We have implemented and tested communication with

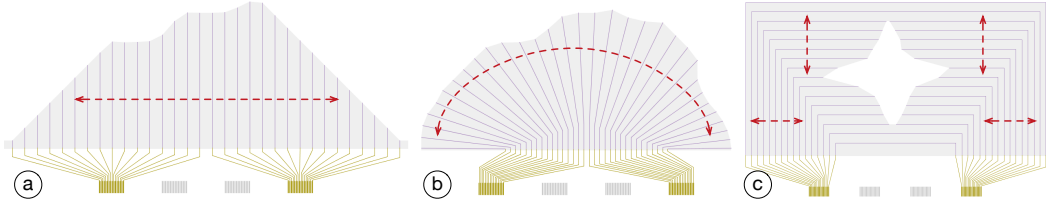


Figure 9: Alternative sensing topologies: (a) A layout of parallel vertical sensors, optimal for horizontal cuts. (b) A star layout, optimal for cuts around the periphery. (c) A topology of double-connected sensors, optimal for vertical cuts and holes. The red arrows show the direction of optimal cuts.

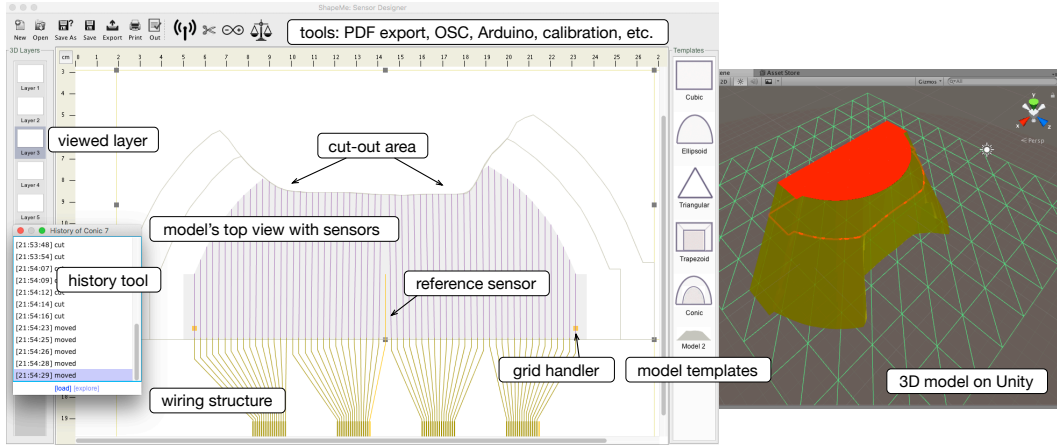


Figure 10: The *ShapeMe* toolkit (left) allows the maker to choose an initial geometry for the physical model and customize the topology of the length-aware sensors. The toolkit also applies a wiring structure to connect the sensors to the *ShapeMe* board. It then receives events from the hardware to update the shape of the virtual model. It keeps a history of the changes. It also communicates with Blender or Unity (right) through OSC messages, where the maker can view the 3D model reconstructed from its individual 2D layers.

both Unity and Blender, where the first allows makers to use *ShapeMe* models in conjunction with AR headsets such as Microsoft HoloLens. Our user interface also provides a history tool to let users review past model edits. A maker can undo unwanted fabrication actions by going back to the history of their actions and then reprint the physical model, in order to follow an alternative fabrication path.

As we saw in the previous section, physical prototyping involves richer tangible manipulations that extend beyond cutting, e.g., folding, rolling, and stretching. Some previous work has developed techniques to support flex or fold sensing [Gon+14]; [VS17]. However, estimating the path of folding interactions with precision that is sufficient for interactive fabrication scenarios is still a difficult problem. An additional challenge is how to support soft and stretchable materials and how to apply visual feedback on their surface. I present our contributions in this direction next.

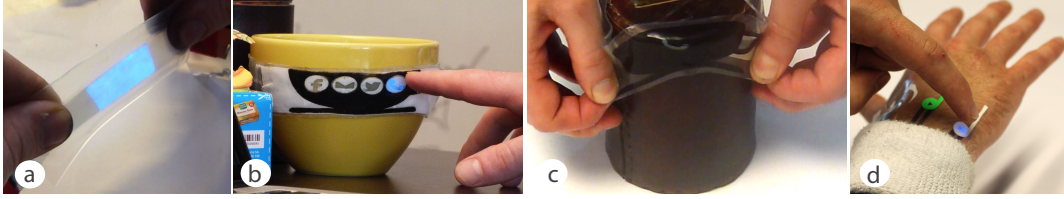


Figure 11: Stretchis [WTM16] are highly stretchable user interfaces that combine touch/proximity sensors and electroluminescent displays (a). Stretchis are transparent (b); can be stretched to fit the geometry of physical objects (c); and can act as on-skin user interfaces (d).

3.4 EMBEDDING INTERACTION IN STRETCHABLE MATERIALS

Stretchable electronics is a very active research area in advanced materials research. A significant body of work has focused on how to apply conductive inks (e.g., graphene, PEDOT:PSS, silver nanowire networks) on silicon-based organic polymers (PDMS) [Kim+09]; [Lip+12]. Other research has investigated stretchable electrochromic [Yan+14] and electroluminescent displays [Wan+15]. A main challenge for HCI research is how to bring the results of such research to wider audiences, and in particular, how to enable casual makers to create interactivity stretchable components as part of their own object designs. *Stretchis* [WTM16] was a first step to this direction. Even though our work was largely inspired by advances in materials science [Lip+12]; [Lu+14]; [Wan+15], our fabrication method does not require expensive equipment to dispose and pattern conductive material. It is therefore better adapted for quick prototyping and is more accessible to non-expert practitioners.

Figure 11 shows examples of stretchable user interfaces (*Stretchis*) fabricated with our method. *Stretchis* use PDMS as the base material for embedding touch and proximity sensors and provide visual output by means of electroluminescent displays. They can be very thin ($\approx 200\mu\text{m}$). Due to the softness of their base material, they can be rolled, folded and stretched to over 100% over their natural length. Hobbyists, designers, or HCI researchers can use these properties to add interactivity to the surface of diverse materials, including fabrics, shape-changing surfaces, and the human skin.

Our fabrication approach is based on *screen printing* (or *serigraphy*), a traditional printing technique used to transfer ink onto a surface (see Figure 12a). The screen-printing technique has been popularized by notable artists. Andy Warhol’s *Marilyn Diptych* portraying Marilyn Monroe in 50 copies, is probably the most famous painting produced with this technique. Interestingly, the same technique is commonly used by advanced materials research to print functional inks (e.g., conductors) for evaluation purposes. Screen printing was first applied in HCI research by Olberding et al. [OWS14]. Michael Wessely had actively participated in this project and introduced the technique to our laboratory when he started his Ph.D. thesis.

Olberding et al. [OWS14] used screen printing to print touch sensors and electroluminescent displays onto diverse materials, including paper, leather, wood, and ceramics. The method interleaves layers of a transparent conductive ink (PEDOT:PSS) and phosphor, which emits light under high voltage (but low current). Unfortunately, the same method does not work for stretchable PDMS substrates.

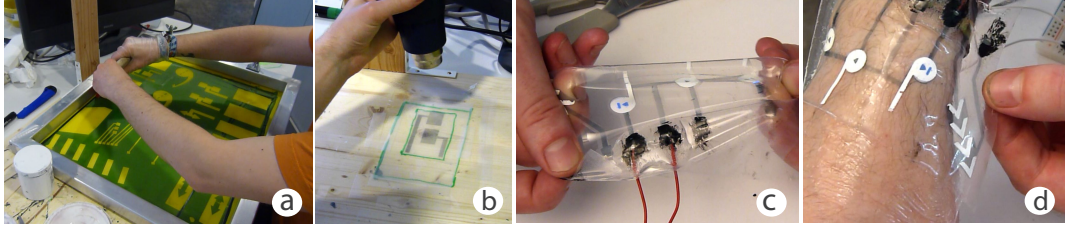


Figure 12: Fabricating stretchable touch sensors and electroluminescent displays [WTM16]. (a) Using a screen printing net (produced with UV lithography) to print layers of functional inks onto a stretchable PDMS substrate. (b) Each printed layer is cured or heat-dried before the subsequent layer is printed on top. (c) The designer adds power and (d) tests the stretchable component on his skin.

The reason is that the conductive ink (PEDOT:PSS) is water based, while PDMS is hydrophobic. As a result, the ink forms drops on the surface of PDMS and stops being conductive.

Previous work on advanced materials research [Lip+12] has shown how to print PEDOT:PSS on PDMS and retain conductivity for up to 188% strain. However, such methods are based on special treatments, e.g., corona and plasma treatments that require specialized equipment, while depending on their type, their effect may be temporary. After experimenting with various treatments, we developed a simple, inexpensive and permanent method. Since direct application of the conductive ink is not possible, we use a stretchable and transparent binding layer as an interface between the PDMS and the ink. This binder contains a lower percentage of water than the PEDOT:PSS ink, which reduces the repellent effect. Also, because it is highly viscous, the ink does not form drops on the PDMS surface. We further adapted Wang et al.'s [Wan+15] fabrication method to produce a stretchable phosphor layer by mixing phosphor particles with fluid PDMS.

The fabrication of *Stretchis* is based on a multi-layer approach that combines four types of functional layers: (1) the base substrate layer that consists of pure PDMS, (2) the sensing layer – conductive ink patterned according to specific input requirements of an application, (3) the display layer that provides electroluminescent visual output as described above, and (4) the aesthetics layer that consists of patterned color inks. This multi-layer approach gives significant freedom to designers, since layers are independent and can be customized and printed during different phases of the fabrication process. Furthermore, layers can be interleaved in many ways, depending on the requirements of a *Stretchis* application. For example, if proximity sensing is important, the sensing layer must be printed on top of other layers for higher precision.

We ran series of tests to evaluate the conductive behavior and endurance of *Stretchis*. We showed that we can reliably detect direct touch and proximity (up to 6 cm away) for strains up to 120% of the natural *Stretchi*'s length. We further showed that *Stretchis* remain conductive even after 6000 stretches with strains up to 50%. We recorded a break point at 153% strain. Such performance was way beyond the state-of-the-art in HCI research at that time. As a reference, the *iSkin* sensor [Wei+15b] had been tested under up to 30% strain.

A major limitation of *Stretchis* was the lack of strain sensing. Unfortunately, subsequent stretches progressively increase the conductor's resistance. Although this increase does not affect the reliability of touch and proximity sensing, it makes strain measurements difficult. Possible solutions to this problem include the use

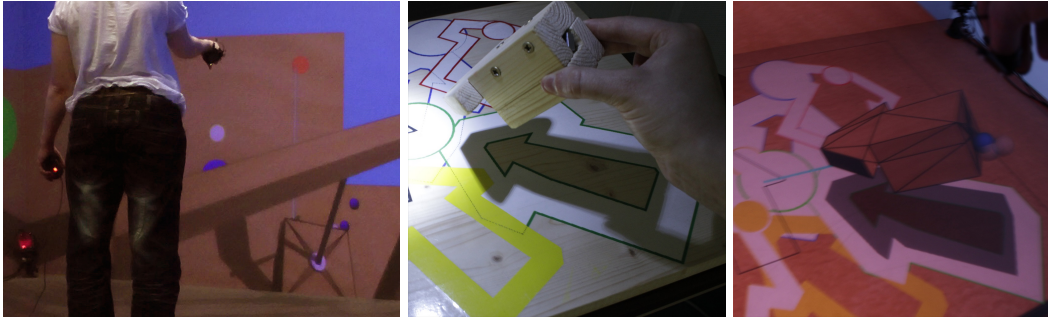


Figure 13: In collaboration with Inria at Sophia Antipolis (Emmanuelle Chapoulie and George Drettakis), we investigated how people manipulate objects in both real and virtual settings [Cha+15]. A user in an immersive (CAVE) environment (left). Completing a 6 DoF manipulation task in real (center) and virtual (right) settings.

of alternative inks, such as silver nanowire (AgNW) networks, which have been shown to support strain sensing [Amj+14], or capacitive sensing through parallel layers of carbon nanotube electrodes [Coh+12]. Strain sensing support offers new application scenarios, such as using *Stretchis* as elastic components of shape-changing objects that sense and react to their deformations. Supporting their fabrication with computational methods and allowing makers to use them in combination with other fabrication materials, e.g., as part of their *ShapeMe* models, is another interesting possibility.

Following our work on *Stretchis*, the HCI community has shown a vivid interest in stretchable interactive devices for casual makers. As representative examples of this line work, I would like to highlight the *Silicon Devices* by Nagels et al. [Nag+18], as they allow for embedding more complex electronics in stretchable casting silicone, and *LASEC* by Groeger and Steimle [GS19], which provides a parametric design process for generating stretchable cut patterns on thin materials.

3.5 CONCLUSION

I discussed how professional and novices design models for 3D objects. Interaction with materials and physical prototypes is an essential part of industrial design practices. But it also helps novices to anticipate fabrication problems, discover solutions, and communicate ideas to peers. A major challenge for HCI research is how to assist this physical interaction and how to support a tighter link between a physical and a digital model. I presented two sensing technologies (*ShapeMe* and *Stretchis*) that contribute to this goal.

Immersive environments promise to make the virtual design experience more and more realistic by supporting direct 3D interaction. However, the need for physical models will not disappear any soon. Several studies have shown the limitations of 3D virtual environments. For example, Jansen et al. [JDF13] have observed that physical touch acts as a cognitive aid when users manually explore physical models – the haptic feedback they provide and their visual realism cannot be faithfully reproduced by virtual environments. Similarly, our own experiments [Cha+15] on 2D and 3D object manipulation (see Figure 13) have shown that physical interfaces tend to perform better than virtual-reality environments, as they do not suffer from lag, sensing disturbances (e.g., due to occlusion), and perceptual deformations.

However, augmented-reality technologies improve rapidly, so there are future opportunities for bringing immersive and physical modeling tools closer together. I discuss such future challenges in Chapter 6.

In the next chapter, I turn my focus to a very different group of creators, composers of contemporary music, but I am still interested in how combining physical and digital representations can support a creative process.

COMPOSING MUSIC

We have seen so far that creators pass from different visual representations (both digital and physical) to reach a final outcome, which can be an illustration or the design of a building, a car, or other physical objects. For other creation domains, where outcomes are not visual artifacts, representations are primarily symbolic and very idiosyncratic. This is the case for many composers of contemporary music, who are the focus of this chapter.

An early study by Letondal and Mackay [LMo7] showed that many composers of contemporary music make extensive use of pen and paper either to sketch early musical ideas or to write their final scores. An interesting observation of this study was that their musical notations can be very personal and are often re-invented at the beginning of each composition project. Often, such notations do not resemble the traditional notations of final scores given to musicians. They consist of early or intermediate languages that allow the composer to express higher-level ideas. For example, they represent complex musical structures that cannot be easily captured with classical notations. In other situations, they describe music transformations whose actual implementation is a computer program. Since music editing software does not support custom musical vocabularies, many composers still use paper to write music. However, computers have a key role in the work of many contemporary composers beyond music editing software. Some composers are keen programmers. Others work with musical assistants, technical experts who assist them in the implementation of their ideas. Visual programming languages, such as OpenMusic¹ and Max², are commonly used by composers and musical assistants to programmatically generate musical sequences through modular visual programs, called *patches*. The notations that composers often develop are parametric representations of such computerized patches. My focus on this chapter is how to extend these music-programming tools with a new space of interaction that supports custom composition representations.

All solutions that I discuss in this chapter are based on technologies of *augmented paper*. Early research on augmented paper dates more than 25 years back [Wel93]; [MP94]. When I started my postdoctoral research, HCI's interest in this area had been revitalized, partly due to the Anoto paper technology³. Anoto relies on a printed but barely visible dot pattern that allows a camera-enhanced pen to detect its precise position on a page. Many HCI groups, in particular at Stanford, CMU, UC San Diego, Autodesk, ETH Zurich, and Darmstadt relied on this technology to develop systems and software toolkits [SKNo7]; [YPRKo8]; [Hei+10] and studied a variety of application domains, from world editing [Lia+08], educational software [Ovi+12], and biology research [TMEo8] to architectural design [Son+09]; [SGHo6] and air-traffic control [Hur+12]. This interest has declined in recent years, partly because of the highly protective business model that companies in this domain have followed, and partly because of the lack of major hardware innovations that could make such technologies popular to a wider audience of users. Although

¹ <https://openmusic-project.github.io/om7/>

² <https://cycling74.com/>

³ <https://www.anoto.com/>

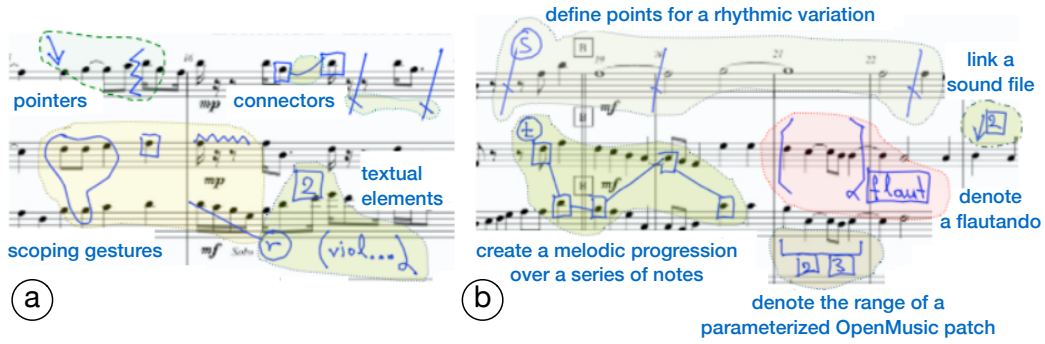


Figure 14: Musink is an extensible gesture-based language: (a) Its syntax is based on a small set of basic, easy-to-recognize gestures: pointers, connectors, scoping gestures, and textual elements. (b) By combining these basic gestures, composers can create their own gesture vocabulary.

it is not clear to me if we must expect a comeback of this technology in the near future, this past research provides great inspiration about how to design interaction for other types of pen input.

4.1 PERSONAL GRAMMARS OF INK GESTURES

A challenge for augmented-paper applications is how to best support the transition between paper and digital content. One of the strengths of paper is the freedom that it gives to its users. Unfortunately, this freedom comes with a cost, as it renders automatic recognition more challenging. A common solution to this problem is to differentiate between handwritten content that may not be recognizable by the system and simple ink gestures that act as system instructions. In most approaches, such gestural vocabularies are close-ended, which simplifies their interpretation by computer applications. A key difficulty of our problem is that composers' vocabularies are open-ended, thus they cannot be pre-specified at design time. Musink [TLM09] is an extensible language of ink gestures (see Figure 14) designed to support such open-ended vocabularies.

Musink bridges physical paper and digital composition tools by letting composers integrate graphical representations of computerized entities into their paper scores. We concentrated on the middle of the creative process, when musical scores are already present, and explored how to augment them with Musink gestures to create new scores. Our design was largely based on interviews of composers and technical experts at IRCAM. Given the prevalence of OpenMusic as a tool at IRCAM, we decided to integrate several aspects of OpenMusic's design philosophy into our approach. Specifically, we treated gestures as functions that can take properties of musical objects as arguments, e.g., their rhythm or pitch, and generate new objects.

We tried to respect the natural role of gestures on traditional paper. Although paper-based gestures can serve as commands that perform software operations, they are also declarative, with a representation designed to be recognizable by humans. We thus studied existing forms of annotations of musical scores [Wino6]; [Cha+07] and integrated them into Musink gestures.

The basic Musink syntax supports three basic elements (see Figure 14a):

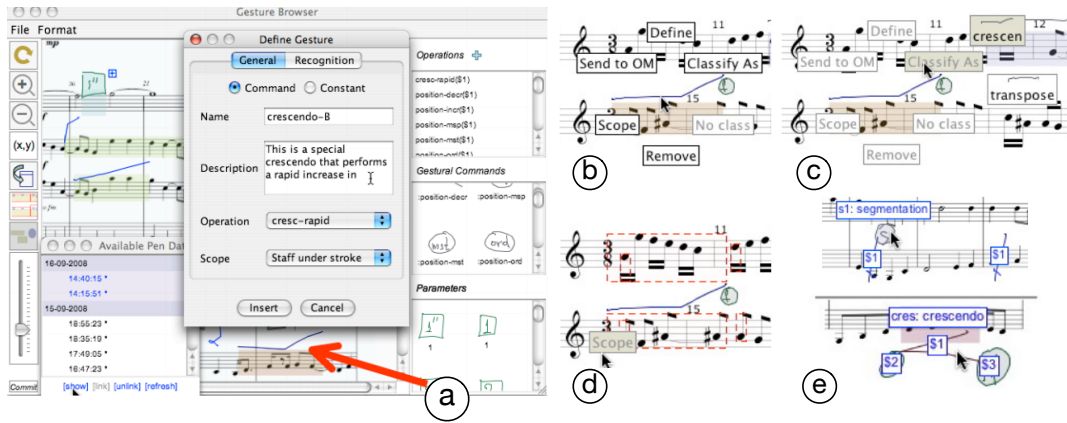


Figure 15: Recognition and interpretation is mediated by Musink’s Gesture Browser. (a) A new gesture is first defined by specifying its identifier, its scope, e.g., time range, pointers in the score, and group of chords, and its numerical or textual parameters. This defines how recognized gestures are translated to computerized functions that external music software can process. (b, c) The user can reinforce or correct the classification of a gesture, (d) refine its scope, and (e) review the recognition of its parameters.

- *Pointers*: They describe specific locations within the score’s timeline. They can take the form of vertical curves or arrows.
- *Scoping Gestures*: They define a range within the score, either as a set of musical symbols or a temporal range. They can take various forms: closed curves, horizontal strokes under or over a staff, and parenthesized scopes defined as in PapierCraft [Lia+08].
- *Text and Parameters*: They can serve as annotations, identifiers or parameters. They are enclosed in parentheses, in a circle or in a rectangle. They may be linked to pointers or scoping gestures if they touch or are close to the gesture.
- *Connectors*: They are supplementary strokes that group elementary gestures together: line segments that visually connect the trace of two gestures or marks indicating a group of traces with a series of small line segments. Those are useful when connected elements are spatially distant.

By combining these basic gestural elements, composers can build their own syntax of gestures. Figure 14b illustrates several examples of complex Musink gestures. This approach aims to optimize the trade-off between openness and recognizability. By separating recognition into multiple steps, it simplifies both recognition and customization. Recognizing elementary gestures is relatively easy, since it involves only a few fixed gestures. However, this small gesture set can produce diverse graphical representations for any given function.

Users can assign semantics to gestures via Musink’s Gesture Browser (see Figure 15). The Gesture Browser lets them define new gestures and refine gesture recognition results. They can remove a basic gesture, revise its recognized scope, define a new gesture class and associate (or disassociate) the gesture with a previously defined class. Gesture classes can function either as gesture identifiers or as parameters. They can also be linked with user-defined OpenMusic functions as they are defined.

Users do not need to have a formal semantic definition of a *Musink* gesture when it is first drawn on paper. The gesture may act solely as a structural element in the score or as a symbol that represents an abstract idea. The user can revisit it later to, for example, assign it semantic meaning or link it to another gesture. We referred to this approach as *semi-structured delayed interpretation*. *Musink* uses identifiers to define semantics. A pointer or scoping gesture may use its own graphical representation as an identifier. For example, the zigzag shape of a horizontal line may act as the identifier for a “tremolo” gesture, distinguishing it from other horizontal lines. Alternatively, a text ‘tag’ may act as an identifier when attached to any pointer or scoping gesture. Any identifier can represent a computer function, e.g., an OpenMusic patch. The function can take any of the following as arguments: score positions, musical symbols, temporal ranges, text and numeric parameters associated with the identifier, either directly or through connectors.

We conducted a series of mini-workshops with individual composers to explore uses of *Musink*. We met with five composers, where four were senior with many years of composition experience. The composers proposed a range of scenarios, such as using *Musink* gestures to control electronics during live performance and composing with symbols (e.g., textual) whose identity or shape controls programmable musical parameters. Several composers emphasized the need for expressive gestures in a score, in addition to symbolic musical representations. We also learned that composers’ gestures on paper are often very precise, representing concrete ideas rather than sketches. Overall, the composers saw in our system a valuable tool for programming music by drawing ink gestures.

Based on our workshops, we extended our tool to better support graphical representations and provided support for paper formats that may not contain common musical notation, e.g., empty staves and graph paper.⁴ However, we also realized that the goal of *Musink* was ambitious. The five composers demonstrated very diverse work approaches and notations that could not be easily accommodated by *Musink*’s syntax. Another limitation of our work at this time was the lack of visual or audio feedback when writing on paper. This lack of feedback made gesture recognition harder and prevented composers from understanding their syntax. In addition, composers could not interact with the paper interface, for example, to quickly explore variations or test ideas. I discuss solutions to these issues in the following sections.

4.2 SUPPORTING CUSTOM COMPOSITION INTERFACES WITH PAPER SUBSTRATES

When Jérémie Garcia started his Ph.D. thesis, we opted for new pen technologies (e.g., Livescribe pens⁵ and bluetooth Anoto pens) that allowed for direct feedback and live communication with a computer. Jérémie completed his thesis in close collaboration with IRCAM. This allowed him to have frequent interactions with composers and musical assistants and conduct a series of studies, including interviews, participatory-design workshops, and longer-term studies that explored the use of early prototypes and technology probes [Hut+03]. Furthermore, he had the chance to work closely for several months with composer Philippe Leroux for the

⁴ A simplified version of *Musink*’s browser for empty staves is available at <https://www.lri.fr/~fanis/omusink/index.html>.

⁵ <https://www.livescribe.com>

creation of his awarded piece *Quid sit musicus ?*⁶ Leroux' composition was covered by the media and was presented in several concerts in France.

Our early studies [Gar+11] investigated how composers can *express* and *explore* musical ideas by using pen and paper to graphically control parameters of their computerized patches. Several composers emphasized the tangible nature of such interfaces. A composer found that interacting with the pen involved a physical movement like playing a musical instrument: “Here, I play the pen.” Other composers appreciated the link between the physical ink left on paper and their computer programs, where hand-written gestures act as *memories of sounds* that can be replayed or further used as guides for future refinements. Another key observation of our early studies was that composers appreciated the power of music-programming environments, such as Max and OpenMusic, and some relied on them, often reusing objects and patches developed for their previous projects. Most agreed that integrating interactive paper interfaces directly into existing computer tools was the correct direction.

Our work on *paper substrates* [Gar+12]; [Gar+14a] takes this approach further by investigating how interactive paper interfaces can extend the interaction vocabulary of music-composition and sound-synthesis tools, such as Max/MSP, OpenMusic, and Ableton Live. Professional musicians use these tools to generate music by programming complex musical objects or mix and arrange sounds. Yet, several contemporary composers find that their interfaces lack support for personal representations that they often use to write music. Figure 16a presents a paper prototype (of an interactive paper interfaces) created by a composer during a participatory-design workshop. According to this scenario, the composer writes music and creates specialization effects for a horn quartet. He uses small, pre-formatted strips of paper to define each parameter of the music (rhythms, pitches, spatialization). Each strip supports a specialized notation, e.g., whole notes for pitches and circular positions for specialization and communicates with *Finale*, a music notation software. The composer organizes the strips on his score page and links them together. He further uses strips of translucent paper to create superimposed layers of musical symbols. This physical form of interaction enables the composer to concentrate on different aspects of the musical sequence (e.g., rhythm vs. spatialization), reuse elements of his music, explore alternative solutions, and easily recombine them in his score. We refer to such modular pieces of interactive paper as *paper substrates*. Notice that the term “substrates” appears frequently in the recent work of Michel Beaudouin-Lafon and Wendy Mackay, as a fundamental concept that complements the *instrument* in Beaudouin-Lafon’s instrumental interaction [BL00]. I will reflect on these concepts in my concluding section.

Figure 16b presents an interactive application that further demonstrates this concept. Its implementation is based on Anoto technology and bluetooth pens that communicate directly with a computer. In this example, the user edits a musical sequence and controls the amplitude of a background sound. The interface consists of a paper substrate that displays a printed (but editable) musical sequence (1), two paper substrates (3, 4) for drawing graphical parameters, and a paper substrate (2) for selecting (by drawing arcs) subsequences in the musical timeline. The user can then play the selections and listen to the result by tapping on the trace of an arc. Paper substrates communicate their data and state to each other. Although they provide a new layer of user interaction, they rely on existing musical software

6 <http://medias.ircam.fr/x93c854>

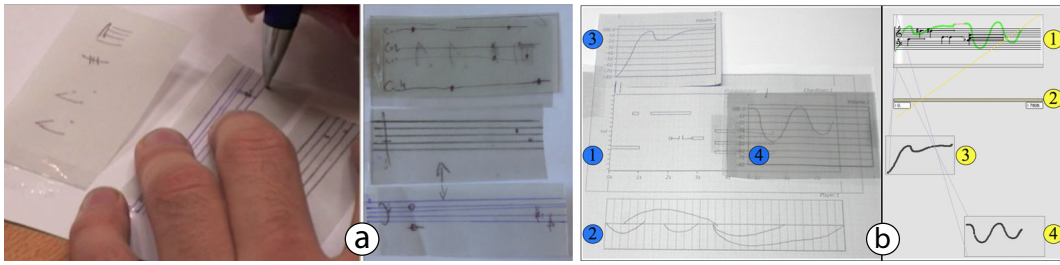


Figure 16: Paper substrates – from participatory design to implementation. (a) A composer proposes a user interface of movable and transparent pieces of paper. Left: The composer draws the pitches of a musical sequence on paper strips with printed staves. The pitch notation is associated with rhythm notation drawn on a different piece of paper. Right: Several linked layers of paper define the final result. Connected elements (either linked with the pen or superimposed) are separate musical objects that share a common timeline. (b) A interactive prototype for editing musical sequences on paper. Left: Paper substrates with printed and handwritten representations of the musical data. Right: Their digital counterparts in the Max/MSP environment.

for processing and storing the actual data. In the above example, each paper substrate is mapped to a unique Max/MSP object, while all links between them are represented by links in the Max/MSP interface.

We met with four composers and a musical assistant to explore uses of paper substrates throughout their interactions with musical data. We observed that paper substrates can take the role of physical proxies of computerized containers of data structures, such as musical sequences, sound signals, and curves. They can also serve as proxies of programmable operators, such as filters and selectors, but also as proxies of programmable modules, e.g, ones that represent higher-level music composition rules. Rather than creating a unique application or a generic user interface, we concentrated on how to provide flexible tools that allow composers or their musical assistants to develop their own paper substrates by assisting reuse.

We developed the PaperSubstrates toolkit [Gar+14a]. The toolkit consists of (i) a Java API that helps developers implement paper substrates with minimal code, and (ii) PaperComposer, an interface builder a customization tool for end-users. The Java API provides support for handling pen events, storing pen data, and recognizing gestures. It also takes care of the management of the Anoto dot pattern, the management of sessions, and the communication of substrates with external applications, such as Max/MSP. As with *Musink*, communication is based on the Open Sound Control (OSC) protocol, where an OSC server allows registered applications to send and receive updates from paper substrates. We support a variety of data types, from basic ones, such as lists of numerical values, to more complex, such as sequences of music notes and chords. Data types are extensible. Developers can add new data types and implement new substrates.

PaperComposer is a graphical user interface for customizing, deploying, and debugging interactive paper applications. Figure 17-Left shows its main screen. PaperComposer organizes paper substrates into printable documents of multiple Anoto pages. The tool enables users to build documents from predefined substrate classes. It relies on Java Reflection for loading new substrate definitions at run time. Users can customize the layout and properties of the substrates and connect them with OSC data channels of external applications (OpenMusic or Max/MSP). They

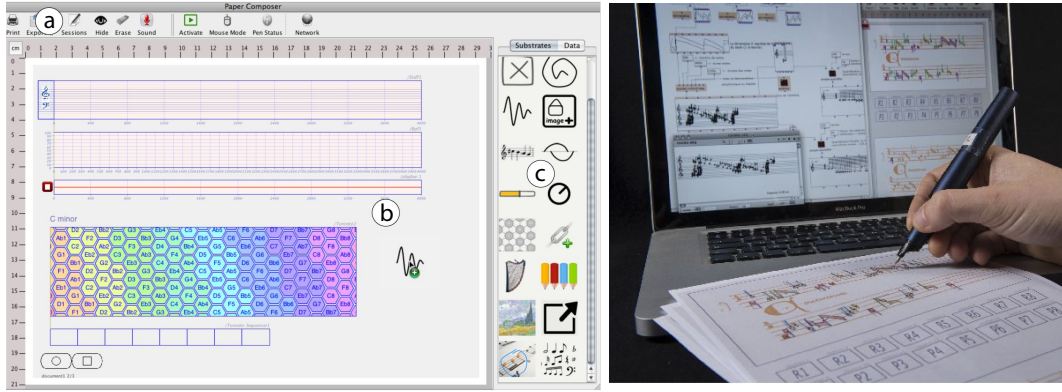


Figure 17: Left: The graphical user interface of PaperComposer. (a) Toolbar with basic functions for managing documents and setting up the parameters of applications. (b) A virtual page with different instances of paper substrates. (c) Thumbnails of available substrate classes. They can be dragged into the virtual page to create new instances. Right: The tool was used for the composition of *Quid sit musicus ?* by Philippe Leroux [GLB14]. Photo by H. Raguét ©Inria.

can also customize their paper substates to communicate with each other through static or dynamic (i.e., created at run time) links.

The toolkit has been used for the development of several applications, including the research tools that I present in the following sections as well as the paper interfaces (see Figure 17-Right) that Philippe Leroux used to produce parts of his piece *Quid sit musicus ?* [GLB14].

4.3 INTERACTIVE RECOGNITION OF HANDWRITTEN INPUT

We saw earlier that *Musink* delays the interpretation of ink gestures. A benefit of this approach is that composers can concentrate on their creative task without having to worry about technology-related problems. For several scenarios, however, high-accuracy recognition is important, and the absence of direct feedback can hamper user performance. For example, many participants of our studies could not anticipate how much freedom the system allowed for and how (or how accurately) automatic recognition worked. In our designs of paper substrates, we tried to reduce recognition problems through well-structured graphical layouts that guide or constrain the entry of notation. We also encouraged users to benefit from the visual and audio feedback provided by the musical environment on the computer. However, this requires users to shift their focus from their workspace on paper to the computer screen to check for errors, often switching between input devices to deal with navigation issues. Unfortunately, interaction can become inefficient and error-prone, especially when users work with musical notation and interleave it with free annotations or specialized (e.g., *Musink*) syntax.

I looked for solutions that could help users to better anticipate and deal with recognition problems by bringing interaction and feedback closer to the area of writing [Tsa12]. Previous work has used interactive devices such as a PDA [Mac+02], a miniature projector [Son+10] or a smartphone [Hei+12] to support interaction with paper. I chose the third approach, as smartphones are widely available, come in different sizes, and can support touch input in addition to visual and audio feedback.

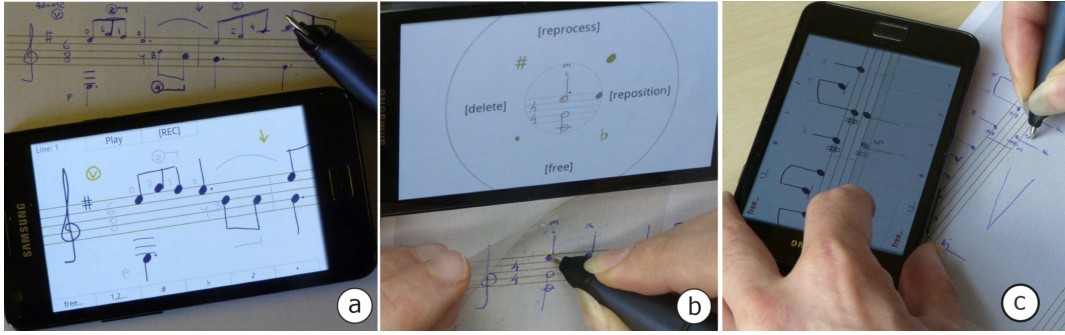


Figure 18: Interactively guiding interpretation while writing musical notation on paper [Tsa12]: (a) A handwritten score as transcribed by the device. (b) Correcting the recognition of a symbol with the pen over a small plastic sheet. (c) Guiding the recognition of strokes with the non-dominant hand.

My design is based on phrasing [Bux95] principles for pen + touch interaction [Hin+10]. My goal was to avoid the use of persistent modes by taking advantage of the kinesthetic coordination of the two hands. In particular, I examined forms of bimanual coordination (see Figure 18) that combine pen and finger touch and make use of the smartphone's touchscreen to control how handwritten data are recognized by the computerized system.

I identified two alternative strategies of user control that aid the interpretation of handwritten data:

- S1. The user relies on the system's ability to correctly recognize handwritten strokes and only intervenes to correct mistakes.
- S2. The user determines how the system should interpret pen strokes while he or she writes on paper.

The success of each strategy depends on how well the system recognizes the strokes and how easy for users it is to interactively direct or correct recognition. It further depends on how well users could foresee the success or failure of the recognizer to optimize their strategies. I studied the two strategies by focusing on three primitive tasks:

- T1. Writing down a stroke or a group of strokes.
- T2. Selecting one or multiple strokes.
- T3. Specifying a meaning for the selected strokes or a context for their interpretation, e.g., a certain vocabulary that constrains their recognition. The user performs this task with a gesture or a menu selection.

The three primitive tasks can be carried out in various combinations, where pen and touch input can overlap in time. T1 is always performed on paper with the pen. T2 and T3 are performed by using either the pen or the mobile touchscreen. I examined four alternative forms of bimanual coordination (*Pen + Pen*, *Pen + Touch*, *Touch + Touch*, and *Touch + Pen*), where the primitive tasks T2 and T3 are executed by either the pen or the touch in different sequence orders. All the four techniques allow for a posteriori error corrections (strategy S1). In addition to error correction, *Touch + Pen* can enforce or constrain how the system will interpret a group of strokes (strategy S2). The *Touch + Pen* technique relies on the activation of a *bezel*

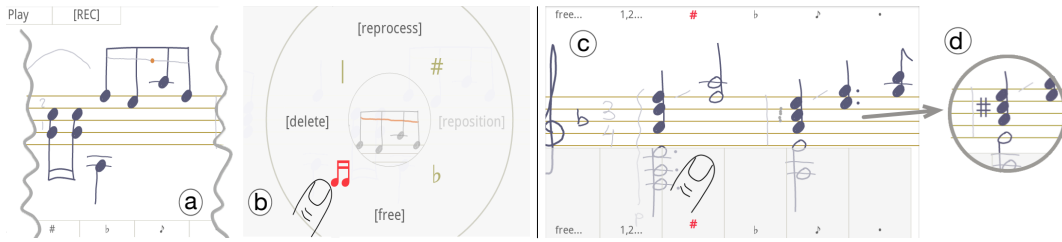


Figure 19: Interacting with musical notation. (a-b) Using *Pen + Touch* to turn an unrecognized horizontal line into a beam. (c-d) *Touch + Pen*: Activating a bezel menu at the bottom of the screen to reinforce the recognition of a sharp.

menu, inspired by the bezel menu of Hinckley et al. [Hin+10] for creating new objects.

I demonstrated the use of the four techniques through an interactive application for writing musical scores. Figure 18a presents an example of a handwritten score and its digitized version on a smartphone. Several strokes that represent key musical elements such as notes, stems, beams and symbols of intonation, e.g., sharps, have been recognized and replaced by dark glyphs. Other elements, such as rests and annotations, remain unrecognized (in light grey). Finally, some strokes represent special functions (in dark yellow), such as audio comments on the score. A design problem is how to represent recognized symbols, as the personal writing style of each user bears recognizable landmarks, essential for human recognition and navigation. Such landmarks are destroyed if the system replaces handwritten symbols by fonts that are commonly used in printed and electronic scores. Keeping or simply highlighting the original form of recognized strokes is not a satisfying solution either, because users need feedback about how the system interprets their handwriting. My solution followed an intermediate approach. Some musical elements such as the head of notes, flats, sharps and keys are replaced by predefined beautified profiles and are positioned with respect to their recognized location in the score structure. Other elements such as stems and beams are only slightly beautified, in a way that their relative geometry in the score is preserved.

I developed a custom online recognizer of musical notation that can be used in combination with the above four techniques. It recognizes notes (whole, half, quarter, eighth, etc.) and chords, ledger lines, dots, beams, flats, sharps, and common keys. Recognition relies on a rule engine that applies a set of syntax rules (recognition, reinforcing, and overwriting rules) to derive the type of a new symbol and its connections with existing score elements, recognized or not. This architecture allows for instrumenting interaction by mapping rules to specific user actions. Figure 19 demonstrates the use of the *Pen + Touch* and the *Touch + Pen* techniques to either correct or reinforce the recognition of musical symbols. The same techniques support functions other than the recognition of musical notation. For example, the user can choose from the top bezel menu to play a segment of the score or create a link to a recording.

I conducted an experiment with 16 participants to evaluate the four techniques under two conditions:

- C1. Recognition was *unpredictable*. Participants had no information about whether the system would succeed in recognizing a stroke. They had to depend on the visual feedback of the mobile device to anticipate an error.

- C2. Recognition was *predictable*. Participants knew beforehand whether recognition would succeed or not, so they could develop a strategy before starting a task.

Results showed that *Touch + Pen* was the fastest technique, irrespective of the condition, i.e., whether recognition errors were predictable or not. Nevertheless, subjective user ratings were not consistent with performance results. *Pen + Pen* and *Pen + Touch* were rated the highest across all evaluation dimensions. It seems that participants underestimated the time required to correct recognition errors and the ability of *Touch + Pen* in saving time. Some participants expressed the opinion that *Touch + Pen* required additional concentration and effort because they had to synchronize both hands at the same time and to frequently shift their attention from paper to the device. Results suggest that participants who consistently used strategy S2 managed to gain a 24 to 30% in speed in comparison to the three other techniques. However, strategy S2 also seems to inflate errors, which may explain why four participants did not use it all and why five other participants mixed it with strategy S1. The fact that *Pen + Touch* was highly preferred to *Touch + Pen* by participants implies that Guiard's [Gui87] principle about the sequential order of the two hands may have exceptions, at least when hands operate in different reference frames, the views are split, and precision is critical for both hands, which means that tasks are not entirely asymmetric.

Overall, this work provides insights about how to design interactive recognition mechanisms for two-hand interaction that combine pen and touch. The mobile interface can be used as companion of paper substrates applications (see previous section) by mediating interaction between the user interface on paper and the computer. Although I focused here on classical musical notation, the same approach could be applied to other vocabularies, including mathematical formulas or *Musink*.

4.4 OBSERVING THE USE OF MUSIC CREATION TOOLS

Our research on music composition tools has been based on interviews with composers, participatory-design workshops, as well as informal evaluations of our prototypes. Such methods helped us understand how composers work and explore design solutions with them. Nevertheless, whether and how composers would use our tools was less clear to us, and the above methods are less appropriate for making such observations. In this section, I will discuss a structured (and partially controlled) observation study in which 12 professional composers and musicians use *Polyphony* [Gar+14b], a constrained music-creation environment, to create a variation of a well-known composition.

Field studies that observe music composition in real conditions are rare exceptions [DT07]; [GLB14] because few artists are willing to dedicate time and experiment with explorative (and thus largely immature) technologies for their professional work. A common research approach for studying phenomena that involve users interacting with technology is to simulate these phenomena in laboratory settings and then observe them under well-controlled conditions. HCI research has long experience with operationalizing interaction phenomena through abstract tasks that evaluate productivity and performance. Assessing creative work is more challenging though, as its outcomes are more open-ended and often cannot be evaluated with objective quantitative measures. Still, as we saw in Chapter 3, we can

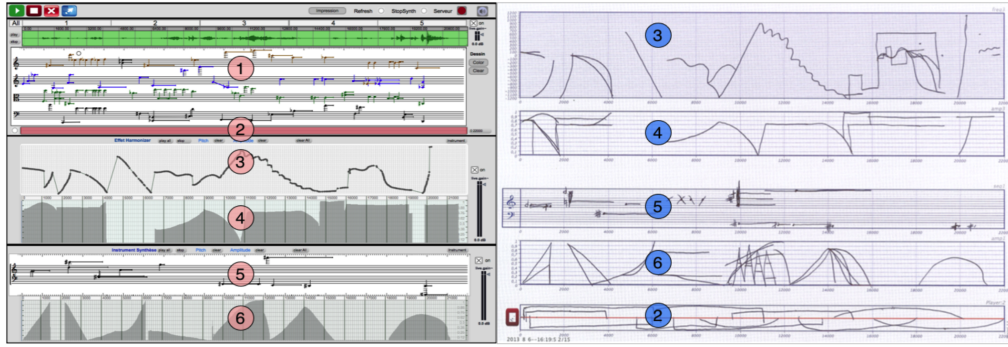


Figure 20: Polyphony’s user interface on the computer screen and on paper, where we show the final score created by a participant [Gar+14b]. Left: Max interface. Right: Interactive paper interface. The top panel displays the waveform and the score transcription of Webern’s piece (1), and a widget to select a time range (2). The middle panel (3, 4) controls the harmonizer effect. The bottom panel (5, 6) controls the synthesizer.

create tasks that simulate the phases of a creative process (object design in this particular case) and systematically observe how participants make use of design tools and materials [Bou+16]. Unfortunately, music composition is a more open-ended and longer process that takes several days, weeks, or even months. No other work has ever examined how this process could be compressed into well-controlled tasks for its systematic study.

The goal of our study was to understand how interactive tools, whether physical or digital, support musical creation, from early paper sketches to the final electronic musical score. In particular, we wanted to explore how and when composers use paper, when they explore ideas by playing on instruments or with the computer, and how they transition between rough ideas to fine-grained representations. Creating an appropriate composition task was our major challenge. We tried to find a task that is short but still creative and meaningful to composers. We worked closely with a young professional composer to design the task and test our first prototypes. The composer was Ph.D. candidate in music composition and had in-depth knowledge of music technology.

The starting point for creating a musical piece is an idea that inspires or drives the creative process. We replaced this phase by a *composition stimulus*, an existing musical piece, that composers could reuse to develop their piece. The composer suggested to use Anton Webern’s Bagatelle No. 2 for String Quartet, Op. 9. This piece lasts only 20 seconds, yet it is widely considered a complete composition. After several iterations with the young composer and a pilot test with a more senior composer, we created a composition task for an audio effect (a harmonizer) and a synthesizer:

“Use the effect to create a variation of Webern’s 20-second piece and write an accompaniment for the synthesizer.”

Although this task does not represent a real-world composition process, it still requires key composition skills. Participants are required to analyze the given material, explore the possibilities offered by the tools, and produce an original musical result.

In addition to the task, we wanted a simple, easy-to-learn interface that integrates key phases of the composition process. Our interface, Polyphony, is a unified

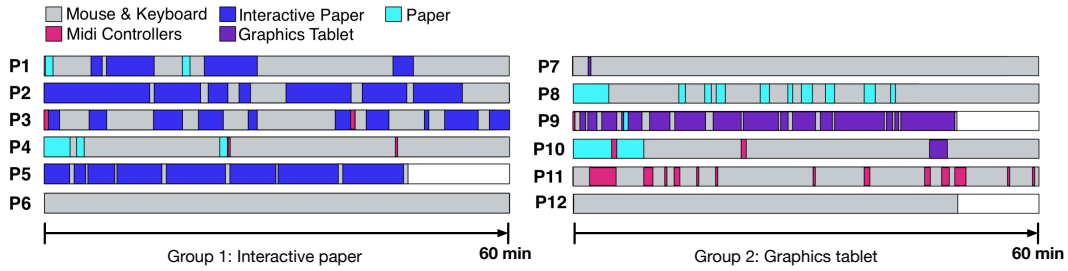


Figure 21: Results from Polyphony's study [Gar+14b]. Use of available inputs and user interfaces.

user interface that provides conventional keyboard- and mouse-based input, pen-based input, as well as a MIDI keyboard for playing the audio effect and the harmonizer. Polyphony supports pen input through interactive paper (i.e., bluetooth pens and Anoto paper) or alternatively, through a graphics tablet. It also includes regular pens and pencils and Webern's original score printed on paper. Figure 20 presents Polyphony's user interface both on the computer and on interactive paper. The computer interface was built within Max. The paper interface was created with PaperComposer [Gar+14a]. Both enable the user to edit and replay the harmonizer effect (pitch transposition and amplitude) and the synthesizer (pitches, durations and amplitude) through interactive components that recognized graphical data or musical notation.

We recruited 12 composers. Ten were professional composers, one was Master student in acousmatic composition, and one was an electronic-music controller engineer. We divided them into two groups. Six participants were exposed to the interactive paper interface, while the other six participants were exposed to a graphics tablet. The participants were given 60 minutes to complete the task. The full procedure lasted 100 to 120 minutes and included an introduction, a training session, and a debriefing.

Figure 21 summarizes the use of the various user interfaces and inputs by our 12 participants. We observe that only four out of the six participants of the first group used the interactive paper interface to work on their piece. The most senior (P4) and most experienced (P6) composers decided to work directly on the computer. They explained that it would be too complex for them to master and produce a satisfying result with the paper interface. In contrast, P5 used the paper interface almost exclusively, while P1, P2, and P3 alternated between interactive paper, mouse and keyboard. In the second group, P9 was the only composer who used the graphics tablet to complete the piece. The composer appreciated the gestural control of the pen, especially for drawing the profile of control curves. A closer look at his score shows curves created with rapid gestures that are hard to reproduce with a mouse. Three other participants briefly experimented with the graphics tablet but quickly abandoned after the first mistakes. The other two composers preferred to use the mouse from the beginning.

We observed a range of interesting uses of Polyphony's paper interface. P1 used pencil to sketch input rhythms, using conventional notation before transcribing them on paper by using the interactive pen. P2 used a ruler with the digital pen to draw precise control lines, while both P2 and P3 printed new pages with their current work state and then re-edited them for more precise results. However, most composers used the mouse to set specific values, such as precise transposition val-

ues in their curves, or to refine pitches that were not correctly recognized by the interactive paper interface. Although some participants (in particular P4, P8, and P10) used pencils and paper to sketch their ideas and annotate content, we observed less sketching than we had expected. A reason given by the composers was that the piece was short so it was easy for them to remember ideas. In addition, as the task was constrained, it did not require the composers to develop a complex esthetic context that might require significant sketching. I should also highlight P11's peculiar composition approach, which consisted in playing the MIDI keyboard along with the audio and then using the mouse and the keyboard to input what he had already played. This composer had no classical training and usually composed music in front of a piano.

We also asked the participants to evaluate and comment on the composition task. All found the task to be interesting and amusing. P1 said that this kind of task *"really helps you think about the impact of electronics on the aesthetics of a piece."* Others considered the task to be a nice composition exercise. We did not evaluate the resulting compositions (e.g., with the help of an external jury), as professionals can be hostile to the evaluation of their personal work. However, we asked them to rate their satisfaction with the final result. All but two participants (P5 and P6) were generally satisfied with their result. P6 (who was also the most experienced composer) would have liked to think more about the musical interest of her piece but would need more advanced tools. P5, on the other hand, was disappointed by the synthesizer's sound.

The Polyphony study has certainly limitations, and it may be hard to generalize its findings. In particular, the tight time constraints discouraged some composers from using devices they were not familiar with. On the other hand, the study was successful in engaging composers in a highly constrained but still challenging task. Several composers explored the potential of the provided tools and appropriated them in ways that provide insights about their strengths and weaknesses. More important, the study provides a unique example for future research on how to design a constrained but creative task in order to study music composition in laboratory settings.

4.5 CONCLUSION

In this chapter, I focused on a peculiar group of creators who work with both paper and computers and develop very diverse representations of music. Our goal was to design tools and techniques that strengthen this diversity of representations. I presented two different approaches. The first (*Musink*) provides a flexible gestural syntax that allows users to define custom representations of computerized patches, which can augment printed musical scores. The automatic recognition of gestural vocabularies on paper can be challenging, but as I showed, it can be guided through interactive devices that offer additional channels of input and output. *Musink* relies on the principle of "delayed interpretation" – the composer may not want to immediately define the meaning or implementation of a gestural phrase expressed with the ink of the pen. Those may be vague and ill-defined at this phase, or simply the composer may not want to interrupt her music-creation process and postpones their actual definition. This principle is powerful with implications for other styles of interaction that do not involve paper (see Chapter 6).

Looking back on this work, I regret for not giving it enough attention in my later projects.

The second solution provides a framework and a set of tools for creating custom interfaces on paper. Such interfaces contain groups of interlinked paper components (paper substrates), where each is specialized to recognize a different notation or a different set of interactions. Paper substrates can be seen as the physical counterparts of computerized objects in visual programming languages. As I mentioned earlier, the notion of substrates has a key role in the more recent work of Michel Beaudouin-Lafon and Wendy Mackay, webstrates [Klo+15] and graphical substrates [Mau+17] being representative examples. Other groups have borrowed this concept. For example, Conversy et al. [Con+18] use graphical substrates as part of a graphical language for authoring airport automations in air-traffic control.

Beaudouin-Lafon [BL17] describes *information substrates* as follows:

“A substrate is a digital computational medium that holds digital information, possibly created by another substrate, applies constraints and transformations to it, reacts to changes in both the information and the substrate, and generates information consumable by other substrates. Substrates are extensible, composable with other substrates, and they can be shared.”

In each of the above systems, the substrate’s role is unique: webstrates [Klo+15] address the sharing of digital content; graphical substrates [Mau+17] focus on the composition and reuse of graphical structures and layouts; paper substrates [Gar+12]; [Gar+14a] deal with diverse representations of musical structures and the recognition of their notations. But the common goal of all these systems is to support diversity and “let users decide which [interaction] style is more appropriate for the situation at hand” [BL17]. As opposed to traditional computer software, where each application (or interface component) has its own closed set of tools and data model, substrates in the above examples may offer a unique representation but refer to a common data model and can link with each other to share their tools. The approach has important benefits. Supporting a new representation (or set of interactive tools) for a user does not require building a new application or changing the software architecture of a system – it is enough to create a new substrate, e.g., by extending existing substrates, which can be used in combination with other substrates and reuse their tools.

Of course, we can also see connections with other parts of my work presented in Chapter 3. Like paper substrates, *ShapeMe* models are physical substrates themselves. Such substrates act on a digital model that is part of a Blender or Unity application, while their constraints are embedded in their sensing topologies. The apparent similarity between the graphical user interface of the *ShapeMe* toolkit (see Figure 10) and PaperComposer (see Figure 17) is not accidental. Both tools share the same software architecture and, theoretically, could co-exist; for example, one could create paper substrates on the surface of *ShapeMe* models to support pen-based model editing like in ModelCraft [Son+06].

Sketch-based languages like *Musink* share many of the goals and properties of substrates, in the sense that they can extend existing user interfaces by offering a new layer of personal representations and interactions. In the next chapter, I examine how such languages can support common non-creative tasks, such as active note taking on paper and data exploration in front of large wall-sized displays.

INTERACTION THROUGH SKETCHING

I discussed roles of sketching for illustration projects (Chapter 2) and design (Chapter 3). Likewise, sketching is a widespread practice in interaction design [Bux07], but HCI research has also tried to integrate sketching capabilities into an interaction vocabulary itself. This line of work goes back to Landay’s seminal work on sketching interfaces [Lang96], while other work has studied sketch-based vocabularies to help users interact through ink gestures [ZMo6]. In Chapter 4, I examined how semi-structured languages based on sketching can provide representational freedom to users, while being interpretable by computers. In this chapter, I continue on this direction but concentrate now on more familiar tasks that do not treat sketches as programmable objects.

I discuss two different examples: *knotty gestures* and *SketchSliders*. Knotty gestures are subtle but distinctive marks sketched on paper that allow users to tag, structure, and interact with their hand-written notes. *SketchSliders* deal with visual-analysis tasks. They are visualization controllers that users freely sketch on a personal tablet to customize their data explorations. Both designs aim to provide personal representations for interaction that support free annotation, bookmarking, and exploration.

5.1 DRAWING INTERACTIVE TRACES ON PAPER

We were interested in exploring ink-based vocabularies for interactive paper notebooks. We started by looking at how biology researchers use pen and paper in connection with data on their computer. As other scientists, several biologists use paper to reflect on data, make calculations on top of them, identify patterns and comment on them. However, biologists also rely on paper notebooks to record their research protocols, data, and results [Mac+02]; [TMEo8]. Clarity and style is important in such scenarios. Our design efforts thus concentrated on vocabularies that could be naturally integrated into a user’s personal writing style, while supporting in-context interaction. These efforts resulted in knotty gestures [TM10]. Knotty gestures aim to balance between expressive power and simplicity and support user interaction with paper (i) at the time of writing and (ii) in the future, when users return to what they have written in the past to rethink and re-interact with it. Although our initial motivation comes from the use of notebooks by scientists, knotty gestures are not specific to this domain. Next, I explain our design concept in more detail.

Knotty gestures are tiny circular gestures drawn on paper that leave a subtle interactive trace, that we call a *knot*. Drawing a knot on paper is a physical interaction that is very familiar to everyone who knows how to write with a pen. More important, the mechanics of this interaction is very distinctive (repeated circling within a tiny radius), leaving a trace that does not easily interfere with other handwritten symbols. Given their size, knots are visible but not obtrusive. Unlike pre-printed paper buttons, which take up space on the paper, they were designed to be easy for the reader to either detect or ignore.

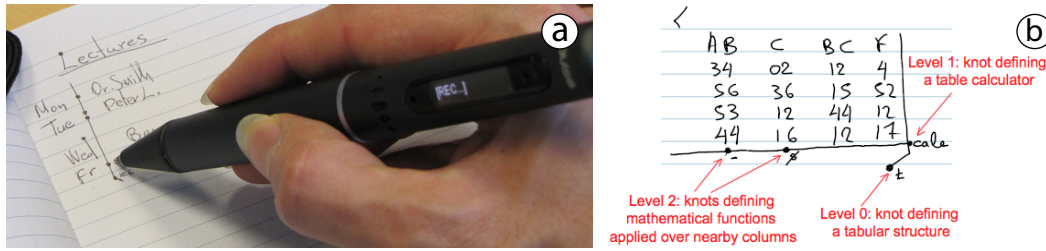


Figure 22: Knotty gestures [TM10]: (a) Drawing knotty gestures over handwritten notes. The knots here, which are dots on lines and characters, activate audio recordings and link them with the notes for future reuse. (b) Nested knots drawn on top of tabular data.

Figure 22a presents a simple example of knots that represent voice annotations. The knots here are drawn on top of a vertical line that groups them together. More generally, knots do not exist on their own. They function as “parasites,” as they are required to reside on top of other handwritten strokes. This design ensures their visual distinction from other similar symbols in handwriting, such as dots in punctuation or bulleted lists. Knots are thus distinguishable and easily recognizable by both humans and computers. We based our implementation on Livescribe pens¹, which supported limited visual feedback (through a 96 x 18 pixels OLED display), a microphone, and a speaker. These pens were equipped with a processor and could be used as independent mobile devices. When we started this project, Livescribe had just launched an open tentative SDK. Its capabilities were very limited at the moment, but it allowed us to program simple pen applications, such as activating mathematical functions over handwritten numbers and adding voice annotations to handwritten notes.

Older systems, such as PapierCraft [Lia+08], make a hard distinction between regular writing and gestures that represent commands. Users rely on mode-switching mechanisms, e.g., pressing a button, which can disrupt the flow of writing. Knotty gestures eliminate the need for mode switching. Unlike pigtails [Hin+05], used by many augmented-paper systems [Lia+08]; [Son+06] as delimiters and command selectors, knotty gestures have a distinct profile that does not interfere with other handwritten symbols. Multiple knotty gestures can be activated on a single stroke and at any position, not only at the end. A knot may also have one or more tails that associate a special function and make it visible to the reader. The lifecycle of a knotty gesture does not end upon its activation, as its trace (i.e., the knot) defines an entry for future interactions. Knots may have local memory, storing their last state of interaction. This state may not be visible, but users can annotate a knot, e.g., with text, to reveal its state.

Inspired by the gestural primitives of *Musink*’s grammar [TLM09], knotty gestures can be combined with other handwritten strokes to build more complex syntactic structures. Like *Musink* identifiers, they can assign functionality to the strokes on which they are drawn. For example, they can convert a line to a link anchor, a list of numerical values, a text selector, or an interactive slider. Alternatively, a knotty gesture can define a scope over a stroke, making it the home of other specialized knots. For example, the vertical line in Figure 22a ends with a tailed knotty gesture that defines the scope of audio annotations. In contrast, the

¹ <https://www.livescribe.com>

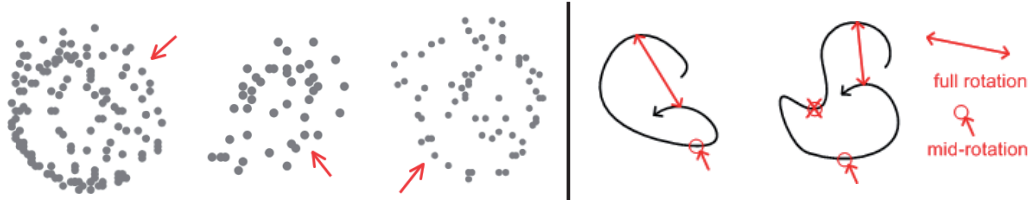


Figure 24: Studying the dynamics of knotty gestures. Left: Examples of knots drawn by different users. The arrows show low-density areas of the knot where rotation is regularly fast. Right: Detection of half and full rotations.

lists. Overall, participants easily learnt how to switch between forward and backward movements through holding and tapping. Circling was less effective though. Several participants complained that they could not easily control the technique, especially when lists were not sorted and had to constantly look at the pen’s display. Designing effective input-control mechanisms for such micro-movements is difficult and requires additional effort.

So far, I have examined how ink gestures can support interaction on paper. Many of the concepts that I presented above can be transferred to pen-based computer interfaces such as pen tablets. For example, Ciolfi Felice et al. [FAM18] have studied the use of knot-based sketches for choreographic annotations on a pen interface for tablets. However, when designing for such devices, the design constraints and interaction possibilities significantly change. I next present a sketching interface that allows data analysis experts to draw personal data exploration widgets and bring part of the data from a large wall-sized display to their personal tablet.

5.2 SKETCHING WIDGETS FOR VISUAL EXPLORATION

High-resolution wall-sized displays have many applications for data analysis, allowing analysts to view, discuss, and visually explore large amounts of data. Interaction in such environments can be difficult, as users are often mobile, moving back to get an overview of their data, and coming up-close to see details [And+11]. Analysts may also need to keep a trace of their explorations for their personal reports and further annotate them to indicate interesting patterns.

In order to address such needs, we introduced a sketching interface (see Figure 25) that splits the data-analysis process between the wall display and a tablet. Users can walk in front of the wall display and interact with its visualizations through *SketchSliders*, free-form range sliders that users draw on the tablet. *SketchSliders* are visualizations by themselves, allowing users to transfer their data explorations to a more personal workspace on the tablet. We support sketched sliders of various shapes and types, including circular sliders for representing periodic data, branched sliders that provide multiple levels of granularity and control, and curvy sliders whose shape can define transformations.

Filtering and range selection are core visual exploration tasks [ASo4]. Traditional visualization systems provide support for these tasks through slider widgets [HS12]. Each slider controls a single data dimension, but users can use multiple sliders to perform more complex queries. Our goal was to combine the simplicity of such widgets with the representation freedom that sketching supports. We were particularly interested in identifying meaningful roles that shaped controllers can take, and explore how users could make use of sketching to augment

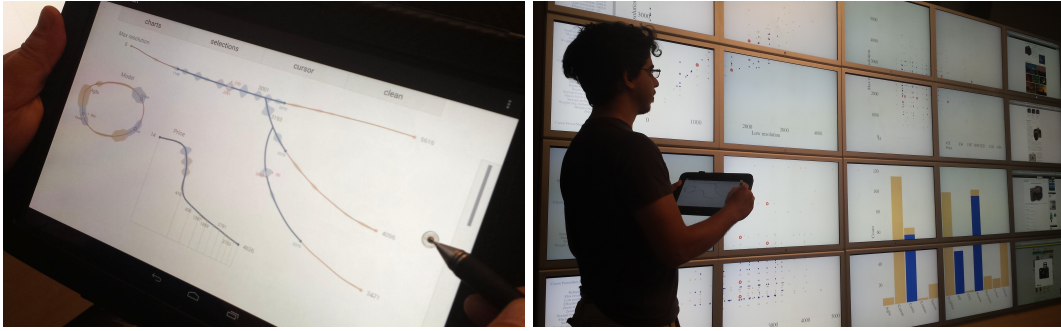


Figure 25: *SketchSliders* (left) allow users to directly sketch visualization controllers to explore multi-dimensional datasets. They can take a range of shapes, including branched and circular, as well as shapes that express transformations. We investigated [TBJ15] how *SketchSliders* can support data exploration in a wall-sized display environment (right), where the queries defined by the sliders are fully synchronized with a visualization dashboard on the wall.

controllers with new filtering mechanisms. We ran individual design sessions with three visualization experts. Our experts suggested that sketching can benefit visual exploration by supporting the following key functions:

Customization. Users can draw arbitrary slider shapes. Such shapes can encode additional information, e.g., a bended slider can indicate a point of interest. A slider’s customized shape can also capture specific data constraints. For example, a circular slider can filter a periodic dimension, e.g., month, with no start and end.

Parametrization. Users can further parametrize their sketched sliders, e.g., by writing values by hand to define slider extremums.

Granularity. Sketching enables users to easily mitigate visual constraints and draw sliders of different lengths, adapted to the level of precision they want. Our experts further proposed sketching mechanisms that allow for variable granularity, such as grafting a slider with long, high-resolution branches.

Annotation and bookmarking. Sketching environments helps analysts to keep a trace of their explorations by augmenting their sliders with bookmarks and personal annotations.

Reusability. The above capabilities facilitate reuse. Sketched controllers do not have to be constantly active. They can temporarily fade out, giving their place to alternative exploration paths but stay accessible for later use.

Our mobile interface integrates many of the above ideas. Users can sketch sliders to explore data dimensions and create queries to filter data (see Figure 25). The sketching interface supports free writing, sketching of interactive widgets, and interaction with gestures, relying on stroke-delimiter techniques [Hin+05] (dwelling and pigtails), gesture recognition, and crossing-based selection [AGo4]. Users combine these techniques to draw sliders, add and manipulate filters, associate data dimensions to sliders, add bookmarks, activate and deactivate sliders, erase widgets, etc. Figure 26 gives a summary of *SketchSliders*’ basic interactions. We support both ratio and ordinal variables (numerical, or textual sorted in alphabetical order).

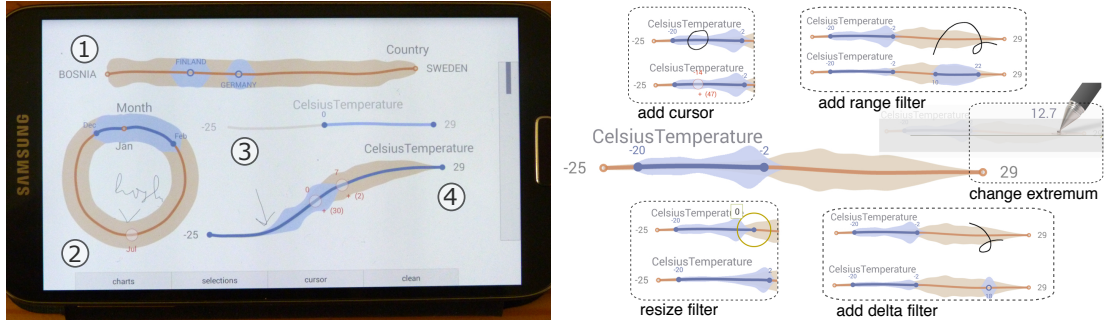


Figure 26: Left: *SketchSliders* on the screen of a small device: (1) a slider over an ordinal variable (Country), (2) a circular slider over a periodic variable (Month), (3) a slider that is currently inactive, and (4) a slider over a ratio variable (Temperature). The query of the three active sliders selects winter mean temperatures for Finland and Germany. Slider 4 shows how the distribution of active temperatures (blue) leans towards lower values. Right: Gestures and interactions to create and manipulate basic slider widgets: add a cursor (crossing circle), add a range or a delta filter (crossing pigtail), change a slider extremum, and resize a filter. Active (blue) density distributions change in response to these actions.

For ratio variables, we differentiate between decimals and integers. We also differentiate between periodic, e.g., months, and non-periodic variables. As shown in Figure 26, periodic variables can be associated with circular sliders that have no ends.

SketchSliders can host multiple interactive filters that define the union of ranges or individual values. We support both range and delta filters, where a delta filter has a single control point and represents either a unique value (ordinal variables), or a small delta range around a value (ratio variables). *SketchSliders* can also host one-dimensional navigation and bookmarking widgets that we call “cursors.” Inspired by scented widgets [WHA07], we augment *SketchSliders* with normalized density distributions. We show both the distribution of the entire dataset (in light orange) and the distribution of its subset (in blue), as defined by the filters of the currently active sliders. Although rough, these visualizations support data exploration directly on the sketching interface. Users can draw multiple sliders to get quick information about how data points are distributed along different dimensions and identify functional dependencies between dimensions as they manipulate the filters of their sliders. Thus, users can focus on their filters without having to shift their attention from the tablet to the wall display.

Finally, we provide two special slider types that allow users to adapt the granularity of their visualizations and their controls. Branched sliders (see Figure 27a) support an infinite nesting of branches. Branches can form arbitrary tree and poly-tree structures and serve as proxies of the main slider. Thus, filters have copies in all slider branches, and the user can choose which copy to manipulate based on its level of precision. By grafting branches and adjusting their extrema, users can zoom in smaller ranges of the data. As they further zoom in, distributions get more fine grained, while filters and cursors become more precise. Branching is especially useful for revealing distribution anomalies and clusters in the data.

Transformation sliders (see Figure 27b) are inspired by how people commonly sketch curves to communicate mathematical functions. We focus on focus+context transformation functions that affect the visualization of the plots on the wall display – peaks of a slider curve represent areas of focus while valleys represent areas

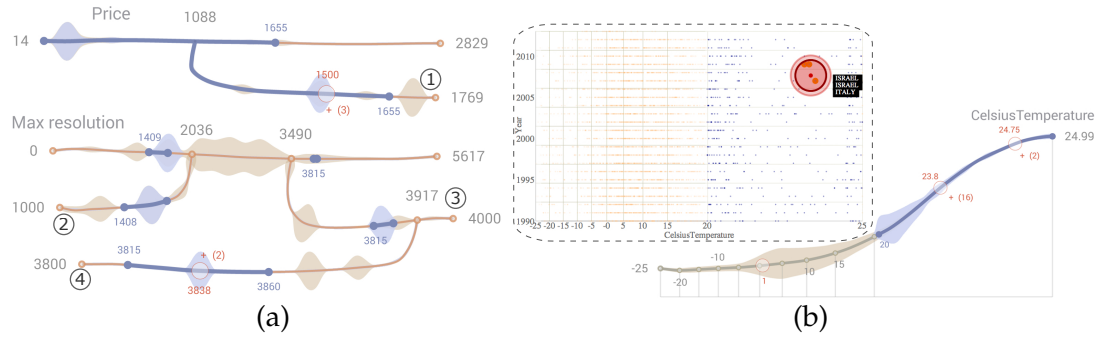


Figure 27: (a) Branched *SketchSliders*. Branches (1, 2, 3, 4) serve as proxies, providing a more detailed view of a range. A branch can start from (1, 3) or end at (2, 4) the main slider or another branch. (b) A Transformation *SketchSlider* applies a fish-eye deformation to a scatterplot. The slider contains grid lines that correspond to the transformed grid of the scatterplot. In this example, the transformation allows for zooming into the range of temperatures between 20° and 25° C. For this range, the user can make more precise selections on the scatterplot and the slider itself.

of context. We define that transformations over a slider path by using a curvilinear $l - y$ coordinate system, where l is the arc length of the partial curve at point $p(x, y)$. This approach overcomes the problem of curves that do not describe valid functions in Cartesian coordinates. As shown in Figure 27b, a transformation applies both to the scatterplot and to the slider itself. This means that values are sparser around peaks and denser around valleys. Similarly, delta filters and cursors become more precise closer to higher peaks.

We ran a user study with six participants (five experienced researchers and one Ph.D. student with expertise in visualization and HCI) to evaluate our system. The participants were seated in front of the wall display and interacted with a 10-inch tablet. They were first given a detailed training session and then performed alone two open exploration tasks. All the participants were very enthusiastic about sketching their own controllers:

“There is something very compelling about sketching your own tools.”

“I can focus either with branches or transformation, you don’t have that in other interfaces.”

They all learned how to *SketchSliders* functionalities but combined a different mix of strategies to complete the tasks. Some participants created branches to increase precision and compare different parts of the dataset (*“I made a second branch to see if I have the same detailed pattern as in the other [branch]”*), while others used transformation sliders to get a closer view of a data range (*“[get] a better view of the densely packed data points in this range”*). All participants found that sketching one controller at a time helped their analysis process. They also liked how the sketching interface was combined with the wall environment and mentioned that the two were *“very well integrated.”* However, participants also commented that *SketchSliders* could be useful in other settings, such as desktop environments. Enabling users to sketch controllers directly on their visualizations is an interesting direction for future work.

5.3 CONCLUSION

I presented two designs of interactive sketch-based UI widgets. Knotty gestures are designed for interactive paper and take into account the constraints of the physical ink. Since ink is permanent, the trace of gestural commands interferes with the ink of the handwritten notes. Knotty gestures offer a clean, non-intrusive method for attaching active annotations to a page. They also provide mechanisms for producing hierarchical structures of semantics by combining knots with hand-drawn line separators or containers of handwritten text. *SketchSliders*, on the other hand, are designed for tablets and thus provide a richer set of gestural interactions and possibilities for dynamic visualization. We saw that *SketchSliders* can also serve as custom data visualizations, moving data exploration from a wall display to a personal space of interaction.

Integrating sketching capabilities into the components of common user interfaces (e.g., a file explorer, an email client, or a statistical analysis tool) is a direction that has been tantalizing me since I started working on *Musink*. Unfortunately, operating systems pose hard constraints on the design and implementation of their graphical user interfaces. Commercial UI widget toolkits have also been very rigid, privileging consistency, structure, and standardization to representational expression. Although I acknowledge that these obstacles are not easy to overcome, I consider them as great opportunities for innovation in HCI research.

CONCLUDING REMARKS AND FUTURE PERSPECTIVES

Goel [Goe14] argues that creative problems require two types of transformations: (i) *lateral transformations*, where solutions diverge from one to slightly different ones, and (ii) *vertical transformations*, where a rough idea progressively converges into a precise and unambiguous version. According to the author, the success of a creative process largely depends on whether the artist’s vocabulary captures the right level of abstraction or precision. Despite their different scope, all systems that I presented in my habilitation thesis target a similar high-level goal: how to support representations and interaction vocabularies that facilitate such transformations. I distill three key lessons from this work:

1. Sketch-based vocabularies can deal with ambiguity and encourage abstraction. In contrast to traditional sketching activities, where a sketch is solely an image subject to the artist’s own interpretation, a computer interface can extract meaning from a sketch, associate with digital content, and make it interactive. In this way, sketches also become tools for structuring ideas and turning them into more precise artifacts or designs. The benefits of interactive sketching go beyond the scope of creative work. Other common tasks, such as data analysis, information management, and decision making, might be better supported by intermediate, ill-defined representations that do not require users to make early commitments.
2. Creators often turn to physical materials because they offer malleable representations and unique aesthetic qualities that traditional digital environments cannot support. I discussed different solutions on how to combine digital creation and material interaction. BricoSketch [TMH15], Musink [TLM09], and Paper Substrates [Gar+12] rely on commercial solutions to support interaction with physical ink, but I also presented examples of custom sensing technologies [WTM18]; [WTM16] that integrate material manipulation into computer-assisted design tools. Arguably, technologies for material-based sensing and interaction are still far from offering the level of precision or robustness that is needed for real creation tasks. However, research on interactive materials is very active [Qam+18], and we can expect that many of the technical limitations of current solutions will be addressed in the future. Furthermore, AR technologies evolve rapidly and offer an additional path in this direction.
3. Computational assistance can benefit both experienced artists and novices. For novices, the key challenge is how to guide the vertical transformation of solutions given that novices lack experience and intuition about how to resolve ambiguities, correct inconsistencies, and move to precise versions. But as many novices also lack basic artistic skills (e.g., drawing), providing alternative paths of creation, e.g., focusing on early prototyping rather than sketching, is important.

For professionals, a first challenge is how to design digital tools that offer the right precision level. Goel [Goe14] reports that designers with deficien-

cies in the right prefrontal cortex generate substandard outcomes because they tend to approach the creative tasks at an “extensively precise, concrete level.” Likewise, a computerized system may hurt the creative tasks if it enforces a vocabulary that is too precise or rigid. A second challenge is how to leverage the practitioners’ personal styles, support their familiar practices and extend, rather than replace, their existing tools. Finally, as professionals are concerned about the speed of their creative process, a third challenge is how to facilitate reuse, assist creation through models, but also reduce the time it takes to switch between different input methods and tools.

I end with a discussion of ongoing work and some future directions that touch the problems that I discussed in the previous chapters.

6.1 DRAWING ASSISTANCE WITH INTERACTIVE REFERENCE MODELS

We saw in Chapter 2 that using photographs as external drawing references is a common practice in professional illustration. An interesting problem is how to best support such practices with tools that combine interaction with automated assistance. Systems such as ShadowDraw [LZC11] and more recently LiveSketch [CBJ19] retrieve photographs from large image collections in interaction with partial user drawings that serve as queries. They then use these photographs as models to assist the drawing process. Despite their algorithmic sophistication, these systems support generic subjects and target users with limited drawing skills. Other systems [HCN19] rely on a dataset of sketch-image pairs to assist UI design based on previous examples of sketches. A common limitation of all these systems is the fact that retrieval depends on global optimization functions. Although the weighting of such functions can be user-defined (e.g., in LiveSketch [CBJ19]), how to map such weights to high-level drawing concepts and embed them in effective interactive controllers is an HCI problem that has not been sufficiently studied.

I am especially interested in image-based approaches that deal with more complex subjects, such as the human body and its postures, by exposing their variations as interactive visual guides. The goal of such guides is to help the artist to constrain the space of outcomes and quickly navigate among possible alternatives. For example, imagine that the artist of Figure 5 interactively specifies her constraints to retrieve photograph models that guide the drawing of hand poses and grasps. Many art books (e.g., see Loomis’ book on figure drawing [Loo43]) describe some very systematic techniques about how to create abstract, expressive models of the human body. This approach poses two main challenges: (i) how to map such abstract models to photograph models, and (ii) how to design interactive tools that allow the artist to effectively control the parameters of these models (e.g., some previous work has examined gestural input that derives expressive characters in motion [GCR13]).

6.2 AR-ASSISTED MODELING

The rapid emergence of affordable VR and AR headsets has considerably facilitated the development of immersive applications, while AR and VR have become the most popular topics of CHI and UIST. Although the significance of such technologies is incontestable for applications in gaming and education, their potentials

for professional creation tasks, such as 3D modeling, is still difficult to assess¹. There is very active research on these technologies that covers a wide range of topics, including immersive drawing [Aro+18], parametric design [Oku+18], and AR-assisted fabrication [Pen+18]. Other interesting work has come up with interaction concepts that support collaboration and the parallel manipulation of virtual objects [Xia+18b].

I have been working with Cédric Fleury and Arthur Fages (Ph.D. student since December 2019) on modeling scenarios for AR, where one or multiple collocated designers work on multiple instances or representations of their models around their common physical space. Our goal is to first better understand some key benefits of hologram-based representations, such as their support for quick physical navigation, their greater visual space without the need of physical screens, and their support for more personal views. We are interested in developing techniques that leverage such potential benefits but also assist collaborators in working together and coordinating their modeling actions.

6.3 SKETCH-BASED INTERACTION VOCABULARIES

Although selection and command specification are two distinct phases in command activation, they commonly appear in close time distance. In tool-based approaches, the user selects the tool that identifies the command and then selects the objects of interest. Pen gesture interfaces often use phrasing techniques that inverse this series of actions [Hin+06]. The user draws one or multiple strokes to select the objects of interest and defines the command immediately after by switching interaction modes. As a key limitation, these approaches require the user to make early decisions about the scope of their intended commands as well as their definition.

Sketch-based vocabularies support persistent tagging mechanisms that divide command activation into separate phases that can appear in any order at any time and can be revisited in the future. As with *Musink*, by combining a set of sketched symbols, one can specify scope, structure (e.g., groups and hierarchies) and functional identifiers that can be later associated with commands, customizable filters, and end-user scripts. My goal is to establish a powerful sketch-based interaction paradigm that brings notions of flexible declarative programming into the user interface itself.

A more ambitious goal is to enable users to sketch interactive controls (interactive links, switches, *SketchSliders*, visualization widgets, etc.) and associate them with their sketched identifiers (tags). For example, suppose a user reads an article and annotates individual words by drawing an identifying gesture, e.g., a zigzag gesture. Later, she draws a circular widget that serves as an index to the article. The user then associates the index with the zigzag gesture to populate it with her annotated words. This interaction approach requires a new set of UI design and development tools. A key problem is that sketches are often hard to parametrize as dynamic objects, e.g., a line sketched over a subset of items in a ordered list may break when the list is reordered. To this end, we need to invent new types of fluid sketch-based UI widgets that support dynamic behavior and effectively deal with their layout constraints.

¹ For an argumentation, you can read Antti Oulasvirta's series of posts: *Nine reasons why I don't believe in current VR/AR technology*: <https://twitter.com/oulasvirta/status/1103298711382380545>

6.4 DESIGNING WITH DATA VS. DESIGNING WITH GRAPHICS

There are other design activities that require working with multiple representations. I have been recently exploring creative tools for data visualization design and infographics [Tsa20]. Design professionals who engage in such activities look for ways to produce compelling graphical representations of data by using a wide range of data analysis and design tools. This problem has recently gained the interest of big research labs such as Microsoft and Adobe Research, and their efforts have resulted in impressive visualization-authoring systems, such as Data Illustrator [Liu+18], DataInk [Xia+18a], and Charticulator [RLB18]. This line of research has been inspired by earlier studies [Big+14] that show that designers prefer a “flexible design environment that does not enforce a specific order of operations” and create visualizations in a “top-down, graphical process,” rather than a bottom-up workflow of visualization grammars and toolkits. The above systems [Liu+18]; [Xia+18a]; [RLB18] support a more flexible design approach through *lazy data binding*, where the design of graphics precedes the data binding step. Even so, these systems still rely on a specific data schema to derive visualization structures. For example, Data Illustrator [Liu+18] requires users to choose a data dimension to repeat a shape at the very beginning of the design process. Other graphics-centric approaches either support simple data structures and no layouts [Kim+17].

My goal is to establish a data-agnostic approach to visualization design that produces reusable visualization structures without programming. My approach combines techniques of graphical properties sharing [HC12], by-example layout specification [RRS13], and persistent alignment and distribution [CF+16] into a new framework of nested property structures. I call it *StructGraphics* (in distinction with “infographics”) to emphasize its focus on graphical structures, rather than data. Figure 28 presents an early version of my prototype. It is composed of three fully synchronized user interface components: (1) the visualization sketcher that allows users to create their graphical structures, (2) a property inspector that exposes all editable graphical properties and automatically organizes them into tabular structures, and (3) a spreadsheet that allows users to create flexible mappings between graphical properties and data at any point in time. Overall, my approach follows the inverse workflow than traditional visualization-design systems. Rather than transforming data dependencies into visualization constraints, it allows users to interactively define the property and layout constraints of their visualization designs and then translate these graphical constraints into alternative data structures.

My longer-term ambition is to turn StructGraphics into a more generic approach that would allow users to turn their sketched representations into data structures and populate them at will with real data. Previous work on grammar-based procedural drawing [Nis+16] provides insights about how to deal with recognition problems when drawing primitives that adhere to a given grammar. Nevertheless, such approaches require users to follow a restrictive, hierarchical sketching workflow that may not reflect how users think and work. I am therefore interested in interactive tools that help users to specify and later refine a grammar as they sketch its graphical primitives.

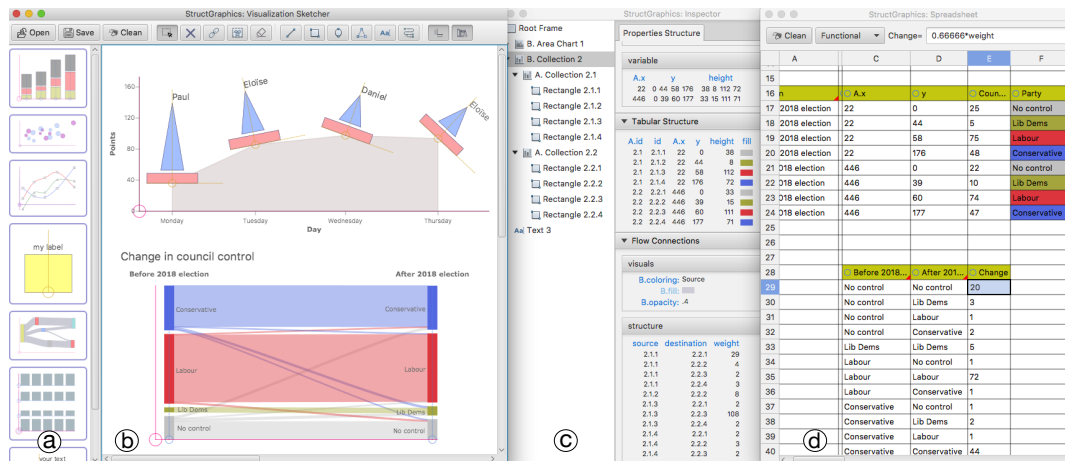


Figure 28: The StructGrapher's user interface consists of: (a) a library of reusable visualization structures, (b) a visualization sketcher for drawing basic shapes and grouping them into collections, (c) a property inspector for exposing and structuring graphical properties, and (d) a spreadsheet for creating mappings between properties and data. The approach enables designers to construct data-agnostic visualizations and create flexible bindings with data. The bottom visualization design was originally published in the Financial Times. See my gallery at: <https://www.lri.fr/~fanis/StructGraphics>

6.5 METHODOLOGICAL QUESTIONS

I close with thoughts about methodological challenges of studying creative tasks. As I discussed earlier, creators often develop idiosyncratic working styles and may express divergent attitudes towards interactive technologies. When presenting our studies on music composers, we were often posed the following frustrating question: "Your results are based on a small handful of very peculiar people. Do they generalize?"

To a great extent, HCI quantitative research focuses on average technology use and average user performance. When the goal is to demonstrate the benefits of a new technique A to a state-of-the-art technique B, a common approach is to take a sample of people, ask them to perform a task with both techniques, and then compare their mean (or median) scores across one or multiple measures. This approach has an obvious limitation. If the investigators fail to demonstrate that average scores of the new technique are superior, they cannot claim any advantage. Yet, the investigators may still observe that their technique brings some clear benefits to some people, although it may be clearly disadvantageous for others.

Unfortunately, group designs are less appropriate in such cases because finding large enough samples of people with common minority patterns is difficult, while confounding variables, such as ordering effects, do not allow for generalizing individual patterns. In my view, the HCI community should turn to *single-case (or single-subject) experiments* [MM09] for many problems where patterns of individual users are of interest. Single-case experiments have a long history in behavioral sciences, although they are less common than group experimental designs. Despite their name, single-case experiments usually involve several participants, e.g., recruiting four or five participants is common in such studies. In contrast to group designs where inferences and conclusions refer to a group, in single-case designs, participants serve as their own controls, and inferences apply to individuals. Of

course, this approach has its own shortcomings. In particular, it requires longer periods of participation such that enough statistical evidence is collected. Finding participants who are willing to participate in long experiments (e.g., with repeated sessions over several days) can be challenging. I am interested in better understanding how single-case experimental methods could benefit HCI research.

Much of the work that I presented in the previous chapters was qualitative in nature. Qualitative research methodologies allow for more flexibility and often deeper and more insightful observations about how creators work. However, the analysis and interpretation of qualitative results is usually subject to the investigators' own convictions and biases. Inter-coder reliability methods help the investigators ensure or at least report on the reliability of their analysis. For example, we used inter-coder reliability measures for the analysis of novice design strategies [Bou+16]. Those helped us to produce a coherent scheme for coding participants' sketches. They also helped us to assess the level of subjectivity of the prototype evaluation measures (e.g., quality and difficulty) that we used. I have further investigated inter-coder reliability measures in the context of gesture elicitation studies [Tsa18], where I found that when authors do not use the appropriate statistics to assess the reliability of their gesture categories, they tend to misinterpret their findings. Specifically, the authors often exaggerate evidence about users' agreement on mappings between gestures and commands.

Yet, inter-coder reliability often requires significant time and resources and may not be practical for certain types of analysis [MSF19]. Other qualitative researchers discourage its use for other reasons. For example, Clarke and Braun [CB16] argue that coding is a *reflexive* process that inevitably bears the mark of the researcher; *"with no one accurate way to code data, the logic behind inter-rater reliability (and multi-independent coders) disappears."* [CB19]. Although I agree that inter-coder reliability can be an overkill in certain situations, I am not convinced by the argumentation of Clarke and Braun. In most practical cases, inter-rater reliability is not used to assess the accuracy of a coding process with respect to a golden standard and does not necessarily restrict the way investigators create their codes or themes. But after the end of a coding process, the researchers still need to verify that their codes can be applied in a consistent manner on the same or different portions of the data by more than one person. Otherwise, the reproducibility of the findings can be put into question. Inter-rater reliability scores should not be simply *"understood as showing that two researchers have been trained to code data in the same way,"* as Clarke and Braun argue [CB19]. In addition to training issues, low inter-rater reliability scores can alarm researchers for other possible problems: (i) codes can be ill-defined, fuzzy, or incoherent; or (ii) the actual data to which the codes apply may be noisy, uncertain, or hard to interpret. Inter-rater reliability scores allow for better assessing the uncertainty of interpretations and conclusions that rely on qualitative results. Future research on qualitative research methods needs to better understand these issues.

BIBLIOGRAPHY

- [AGo4] Georg Apitz and François Guimbretière. “CrossY: A Crossing-based Drawing Application.” In: *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology*. UIST ’04. Santa Fe, NM, USA: ACM, 2004, pp. 3–12. ISBN: 1-58113-957-8. DOI: [10.1145/1029632.1029635](https://doi.org/10.1145/1029632.1029635). URL: <http://doi.acm.org/10.1145/1029632.1029635>.
- [Amj+14] Morteza Amjadi, Aekachan Pichitpajongkit, Sangjun Lee, Seunghwa Ryu, and Inkyu Park. “Highly Stretchable and Sensitive Strain Sensor Based on Silver Nanowire–Elastomer Nanocomposite.” In: *ACS Nano* 8.5 (2014), pp. 5154–5163. DOI: [10.1021/nn501204t](https://doi.org/10.1021/nn501204t).
- [And+11] Christopher Andrews, Alex Endert, Beth Yost, and Chris North. “Information Visualization on Large, High-resolution Displays: Issues, Challenges, and Opportunities.” In: *Information Visualization* 10.4 (2011), pp. 341–355. DOI: <http://dx.doi.org/10.1177/1473871611415997>.
- [Aro+18] Rahul Arora, Rubaiat Habib Kazi, Tovi Grossman, George Fitzmaurice, and Karan Singh. “SymbiosisSketch: Combining 2D & 3D Sketching for Designing Detailed 3D Objects in Situ.” In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI ’18. Montreal QC, Canada: ACM, 2018, 185:1–185:15. ISBN: 978-1-4503-5620-6. DOI: [10.1145/3173574.3173759](https://doi.org/10.1145/3173574.3173759). URL: <http://doi.acm.org/10.1145/3173574.3173759>.
- [ASo4] Robert Amar and John Stasko. “A Knowledge Task-Based Framework for Design and Evaluation of Information Visualizations.” In: *Proceedings of the IEEE Symposium on Information Visualization*. INFOVIS ’04. Washington, DC, USA: IEEE Computer Society, 2004, pp. 143–150. ISBN: 0-7803-8779-3. DOI: [10.1109/INFOVIS.2004.10](https://doi.org/10.1109/INFOVIS.2004.10). URL: <http://dx.doi.org/10.1109/INFOVIS.2004.10>.
- [AWBo3] Saeema Ahmed, Ken Wallace, and Lucienne T. Blessing. “Understanding the differences between how novice and experienced designers approach design tasks.” In: *Research in Engineering Design* 14.1 (2003). DOI: <http://dx.doi.org/10.1007/s00163-002-0023-z>.
- [Bal+99] Ravin Balakrishnan, George Fitzmaurice, Gordon Kurtenbach, and William Buxton. “Digital Tape Drawing.” In: *Proceedings of the 12th Annual ACM Symposium on User Interface Software and Technology*. UIST ’99. Asheville, North Carolina, USA: ACM, 1999, pp. 161–169. ISBN: 1-58113-075-9. DOI: [10.1145/320719.322598](https://doi.org/10.1145/320719.322598). URL: <http://doi.acm.org/10.1145/320719.322598>.
- [Big+14] Alex Bigelow, Steven Drucker, Danyel Fisher, and Miriah Meyer. “Reflections on How Designers Design with Data.” In: *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces*. AVI ’14. Como, Italy: ACM, 2014, pp. 17–24. ISBN: 978-1-4503-2775-6. DOI: [10.1145/2598153.2598175](https://doi.org/10.1145/2598153.2598175). URL: <http://doi.acm.org/10.1145/2598153.2598175>.

- [BL00] Michel Beaudouin-Lafon. "Instrumental interaction: an interaction model for designing post-WIMP user interfaces." In: *CHI*. Ed. by Thea Turner and Gerd Szwillus. ACM, 2000, pp. 446–453. ISBN: 1-58113-216-6. URL: <http://dblp.uni-trier.de/db/conf/chi/chi2000.html#Beaudouin-Lafon00>.
- [BL17] Michel Beaudouin-Lafon. "Towards Unified Principles of Interaction." In: *Proceedings of the 12th Biannual Conference on Italian SIGCHI Chapter*. CHIItaly '17. Cagliari, Italy: ACM, 2017, 1:1–1:2. ISBN: 978-1-4503-5237-6. DOI: [10.1145/3125571.3125602](https://doi.org/10.1145/3125571.3125602). URL: <http://doi.acm.org/10.1145/3125571.3125602>.
- [BM00] Margot Brereton and Ben McGarry. "An Observational Study of How Objects Support Engineering Design Thinking and Communication: Implications for the Design of Tangible Media." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '00. The Hague, The Netherlands: ACM, 2000, pp. 217–224. ISBN: 1-58113-216-6. DOI: [10.1145/332040.332434](https://doi.org/10.1145/332040.332434). URL: <http://doi.acm.org/10.1145/332040.332434>.
- [BM17] P. Baudisch and S. Mueller. *Personal Fabrication*. Foundations and Trends(r) in Human-Computer Interaction. Now Publishers, 2017. ISBN: 9781680832587. URL: <https://books.google.fr/books?id=3T2lswEACAAJ>.
- [Bou+16] Adrien Bousseau, Theophanis Tsandilas, Lora Oehlberg, and Wendy E. Mackay. "How Novices Sketch and Prototype Hand-Fabricated Objects." In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. CHI '16. San Jose, California, USA: ACM, 2016, pp. 397–408. ISBN: 978-1-4503-3362-7. DOI: [10.1145/2858036.2858159](https://doi.org/10.1145/2858036.2858159). URL: <http://doi.acm.org/10.1145/2858036.2858159>.
- [Bra03] B. Bradley. *Drawing people: how to portray the clothed figure*. North Light Books, 2003.
- [Bux+00] William Buxton, George Fitzmaurice, Ravin Balakrishnan, and Gordon Kurtenbach. "Large Displays in Automotive Design." In: *IEEE Computer Graphics and Applications* 20.04 (2000), pp. 68–75. ISSN: 0272-1716. DOI: [10.1109/38.851753](https://doi.org/10.1109/38.851753).
- [Bux07] Bill Buxton. *Sketching User Experiences: Getting the Design Right and the Right Design*. First. Morgan Kaufmann, 2007.
- [Bux95] William A. S. Buxton. "Human-computer Interaction." In: ed. by Ronald M. Baecker, Jonathan Grudin, William A. S. Buxton, and Saul Greenberg. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1995. Chap. Chunking and Phrasing and the Design of Human-computer Dialogues, pp. 494–499. ISBN: 1-55860-246-1. URL: <http://dl.acm.org/citation.cfm?id=212925.212970>.
- [CB16] Victoria Clarke and Virginia Braun. *Thematic Analysis*. SAGE Publications, Inc., 2016. Chap. 6. ISBN: 9781446273753.
- [CB19] Victoria Clarke and Virginia Braun. *Answers to frequently asked questions about thematic analysis*. 2019.

- [CBJ19] John P. Collomosse, Tu Bui, and Hailin Jin. "LiveSketch: Query Perturbations for Guided Sketch-Based Visual Search." In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. 2019, pp. 2879–2887.
- [CF+16] Marianela Ciolfi Felice, Nolwenn Maudet, Wendy E. Mackay, and Michel Beaudouin-Lafon. "Beyond Snapping: Persistent, Tweakable Alignment and Distribution with StickyLines." In: *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. UIST '16. Tokyo, Japan: ACM, 2016, pp. 133–144. ISBN: 978-1-4503-4189-9. DOI: [10.1145/2984511.2984577](https://doi.org/10.1145/2984511.2984577). URL: <http://doi.acm.org/10.1145/2984511.2984577>.
- [Cha+07] Y. Chapuis, D. Fober, S. Letz, Y. Orlarey, and C. Daudin. "Annotation de partitions musicales dynamiques." In: *Actes des Journées d'Informatique Musicale JIM'07 - Lyon*. Ed. by Grame. 2007, pp. 18–27.
- [Cha+15] Emmanuelle Chapoulie, Theophanis Tsandilas, Lora Oehlberg, Wendy Mackay, and George Drettakis. "Finger-Based Manipulation in Immersive Spaces and the Real World." In: *Proceedings of IEEE 3DUI*. 2015. URL: <http://www-sop.inria.fr/reves/Basilic/2015/CTOMD15>.
- [Coh+12] Daniel J. Cohen, Debkishore Mitra, Kevin Peterson, and Michel M. Maharbiz. "A Highly Elastic, Capacitive Strain Gauge Based on Percolating Nanotube Networks." In: *Nano Letters* 12.4 (2012). PMID: 22409332, pp. 1821–1825. DOI: [10.1021/nl204052z](https://doi.org/10.1021/nl204052z). eprint: <https://doi.org/10.1021/nl204052z>. URL: <https://doi.org/10.1021/nl204052z>.
- [Con+18] Stéphane Conversy et al. "Vizir: A Domain-Specific Graphical Language for Authoring and Operating Airport Automations." In: *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*. UIST '18. Berlin, Germany: Association for Computing Machinery, 2018, 261–273. ISBN: 9781450359481. DOI: [10.1145/3242587.3242623](https://doi.org/10.1145/3242587.3242623). URL: <https://doi.org/10.1145/3242587.3242623>.
- [Cro04] Nigel Cross. "Expertise in design: an overview." In: *Design Studies* 25.5 (2004), pp. 427–441. ISSN: 0142-694X. DOI: <http://dx.doi.org/10.1016/j.destud.2004.06.002>. URL: <http://www.sciencedirect.com/science/article/pii/S0142694X04000316>.
- [DHK09] Steven P. Dow, Kate Heddleston, and Scott R. Klemmer. "The Efficacy of Prototyping Under Time Constraints." In: *Proceedings of the Seventh ACM Conference on Creativity and Cognition*. C&C '09. Berkeley, California, USA: ACM, 2009, pp. 165–174. ISBN: 978-1-60558-865-0. DOI: [10.1145/1640233.1640260](http://doi.acm.org/10.1145/1640233.1640260). URL: <http://doi.acm.org/10.1145/1640233.1640260>.
- [Dod85] Bert Dodson. *Keys to drawing*. North Light Books, 1985.
- [Dow+11] Steven Dow, Julie Fortuna, Dan Schwartz, Beth Altringer, Daniel Schwartz, and Scott Klemmer. "Prototyping Dynamics: Sharing Multiple Designs Improves Exploration, Group Rapport, and Results." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '11. Vancouver, BC, Canada: ACM, 2011, pp. 2807–2816. ISBN: 978-1-4503-0228-9. DOI: [10.1145/1978942.1979359](https://doi.org/10.1145/1978942.1979359). URL: <http://doi.acm.org/10.1145/1978942.1979359>.

- [DPH10] D. Dixon, M. Prasad, and T. Hammond. "iCanDraw: Using sketch recognition and corrective feedback to assist a user in drawing human faces." In: *Proc. of the International Conference on Human Factors in Computing Systems (CHI)*. ACM. 2010.
- [DT07] Nicolas Donin and Jacques Theureau. "Theoretical and methodological issues related to long term creative cognition: The case of musical composition." In: *Cognition Technology and Work* 9 (Oct. 2007), pp. 233–251. DOI: [10.1007/s10111-007-0082-z](https://doi.org/10.1007/s10111-007-0082-z).
- [Eck+12] Claudia Eckert, Alan Blackwell, Martin Stacey, Christopher Earl, and Luke Church. "Sketching across design domains: Roles and formalities." In: *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 26 (Special Issue 03 Aug. 2012), pp. 245–266. ISSN: 1469-1760. DOI: [10.1017/S0890060412000133](https://doi.org/10.1017/S0890060412000133).
- [Edw79] Betty Edwards. *Drawing on the right side of the brain*. Tarcher, 1979.
- [EMBS14] Ozgur Eris, Nikolas Martelaro, and Petra Badke-Schaub. "A comparative analysis of multimodal communication during design sketching in co-located and distributed environments." In: *Design Studies* 35.6 (2014), pp. 559–592. DOI: [10.1016/j.destud.2014.04.002](https://doi.org/10.1016/j.destud.2014.04.002).
- [FAM18] Marianela Ciolfi Felice, Sarah Fdili Alaoui, and Wendy E. Mackay. "Knotation: Exploring and Documenting Choreographic Processes." In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, CHI 2018, Montreal, QC, Canada, April 21-26, 2018*. ACM, 2018, p. 448. DOI: [10.1145/3173574.3174022](https://doi.org/10.1145/3173574.3174022). URL: <https://doi.org/10.1145/3173574.3174022>.
- [FBY14] Daniela Faas, Qifang Bao, and Maria C. Yang. "Preliminary Sketching and Prototyping: Comparisons in Exploratory Design-and-Build Activities." In: *ASME Design Engineering Technical Conferences*. 2014. DOI: [10.1115/DETC2014-34928](https://doi.org/10.1115/DETC2014-34928).
- [FGF11] J. Fernquist, T. Grossman, and G. Fitzmaurice. "Sketch-sketch revolution: an engaging tutorial system for guided sketching and application learning." In: *ACM Symp. on User Interface Software and Technology (UIST)*. 2011, pp. 373–382.
- [Fol+10] Sean Follmer, David Carr, Emily Lovell, and Hiroshi Ishii. "Copy-CAD: Remixing Physical Objects with Copy and Paste from the Real World." In: *Adjunct Proceedings of the 23Nd Annual ACM Symposium on User Interface Software and Technology*. UIST '10. New York, New York, USA: ACM, 2010, pp. 381–382. ISBN: 978-1-4503-0462-7. DOI: [10.1145/1866218.1866230](https://doi.org/10.1145/1866218.1866230). URL: <http://doi.acm.org/10.1145/1866218.1866230>.
- [FP16] Jean-Daniel Fekete and Romain Primet. "Progressive Analytics: A Computation Paradigm for Exploratory Data Analysis." In: *CoRR abs/1607.05162* (2016). URL: <http://arxiv.org/abs/1607.05162>.
- [Gar+11] Jérémie Garcia, Theophanis Tsandilas, Carlos Agon, and Wendy Mackay. "InkSplorer: Exploring Musical Ideas on Paper and Computer." In: *New Interfaces for Musical Expression (NIME 2011)*. Oslo, Norway, May 2011, pp. 361–366. URL: <https://hal.inria.fr/inria-00600083>.

- [Gar+12] Jérémie Garcia, Theophanis Tsandilas, Carlos Agon, and Wendy Mackay. "Interactive Paper Substrates to Support Musical Creation." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '12. Austin, Texas, USA: ACM, 2012, pp. 1825–1828. ISBN: 978-1-4503-1015-4. DOI: [10.1145/2207676.2208316](https://doi.org/10.1145/2207676.2208316). URL: <http://doi.acm.org/10.1145/2207676.2208316>.
- [Gar+14a] Jérémie Garcia, Theophanis Tsandilas, Carlos Agon, and Wendy Mackay. "PaperComposer: Creating Interactive Paper Interfaces for Music Composition." In: *IHM'14, 26e conférence francophone sur l'Interaction Homme-Machine*. Lille, France: ACM, Oct. 2014, pp. 1–8. DOI: [10.1145/2670444.2670450](https://hal.archives-ouvertes.fr/hal-01089463). URL: <https://hal.archives-ouvertes.fr/hal-01089463>.
- [Gar+14b] Jérémie Garcia, Theophanis Tsandilas, Carlos Agon, and Wendy E. Mackay. "Structured Observation with Polyphony: a Multifaceted Tool for Studying Music Composition." In: *DIS'14 - ACM conference on Designing Interactive Systems*. ACM. Vancouver, Canada, June 2014, pp. 199–208. URL: <https://hal.inria.fr/hal-00988725>.
- [Gar14] Jérémie Garcia. "Supporting music composition with interactive paper." Theses. Université Paris Sud - Paris XI, June 2014. URL: <https://tel.archives-ouvertes.fr/tel-01056992>.
- [GCR13] Martin Guay, Marie-Paule Cani, and Rémi Ronfard. "The Line of Action: An Intuitive Interface for Expressive Character Posing." In: *ACM Trans. Graph.* 32.6 (Nov. 2013), 205:1–205:8. ISSN: 0730-0301. DOI: [10.1145/2508363.2508397](http://doi.acm.org/10.1145/2508363.2508397). URL: <http://doi.acm.org/10.1145/2508363.2508397>.
- [GGCG17] Neil Gershenfeld, Alan Gershenfeld, and Joel Cutcher-Gershenfeld. *Designing Reality: How to Survive and Thrive in the Third Digital Revolution*. New York, NY, USA: Basic Books, Inc., 2017. ISBN: 0465093477, 9780465093472.
- [GK15] Elisa Giaccardi and Elvin Karana. "Foundations of Materials Experience: An Approach for HCI." In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. CHI '15. Seoul, Republic of Korea: ACM, 2015, pp. 2447–2456. ISBN: 978-1-4503-3145-6. DOI: [10.1145/2702123.2702337](http://doi.acm.org/10.1145/2702123.2702337). URL: <http://doi.acm.org/10.1145/2702123.2702337>.
- [GLB14] Jérémie Garcia, Philippe Leroux, and Jean Bresson. "pOM: Linking Pen Gestures to Computer-Aided Composition Processes." In: *40th International Computer Music Conference (ICMC) joint with the 11th Sound & Music Computing conference (SMC)*. Athenes, Greece, Sept. 2014. URL: <https://hal.inria.fr/hal-01053295>.
- [Goe14] Vinod Goel. "Creative brains: designing in the real world†." In: *Frontiers in Human Neuroscience* 8 (2014), p. 241. ISSN: 1662-5161. DOI: [10.3389/fnhum.2014.00241](https://www.frontiersin.org/article/10.3389/fnhum.2014.00241). URL: <https://www.frontiersin.org/article/10.3389/fnhum.2014.00241>.
- [Gog+18] Anna Gogolou, Theophanis Tsandilas, Themis Palpanas, and Anastasia Bezerianos. "Comparing Similarity Perception in Time Series Visualizations." In: *IEEE Transactions on Visualization and Computer Graphics*.

- TVCG 2019 (InfoVis 2018) 25.1 (Oct. 2018), pp. 523–533. URL: <https://hal.inria.fr/hal-01845008>.
- [Gog+19] Anna Gogolou, Theophanis Tsandilas, Themis Palpanas, and Anastasia Bezerianos. “Progressive Similarity Search on Time Series Data.” In: *International Workshop on Big Data Visual Exploration and Analytics (BigVis), in conjunction with the International Conference on Extending Database Technology (EDBT)*. Lisbon, Portugal, Mar. 2019. URL: <https://hal.inria.fr/hal-02103998>.
- [Gog+20] Anna Gogolou, Theophanis Tsandilas, Karima Echihabi, Anastasia Bezerianos, and Themis Palpanas. “Data Series Progressive Similarity Search with Probabilistic Quality Guarantees.” In: *International Conference on Management of Data (ACM SIGMOD)*. 2020, pp. 1857–1873. DOI: [10.1145/3318464.3389751](https://doi.org/10.1145/3318464.3389751). URL: <https://hal.archives-ouvertes.fr/hal-02560760/>.
- [Gon+14] Nan-Wei Gong, Jürgen Steimle, Simon Olberding, Steve Hodges, Nicholas Edward Gillian, Yoshihiro Kawahara, and Joseph A. Paradiso. “PrintSense: A Versatile Sensing Technique to Support Multimodal Flexible Surface Interaction.” In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’14. Toronto, Ontario, Canada: ACM, 2014, pp. 1407–1410. ISBN: 978-1-4503-2473-1. DOI: [10.1145/2556288.2557173](https://doi.org/10.1145/2556288.2557173). URL: <http://doi.acm.org/10.1145/2556288.2557173>.
- [Gro+02] Tovi Grossman, Ravin Balakrishnan, Gordon Kurtenbach, George Fitzmaurice, Azam Khan, and Bill Buxton. “Creating Principal 3D Curves with Digital Tape Drawing.” In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’02. Minneapolis, Minnesota, USA: ACM, 2002, pp. 121–128. ISBN: 1-58113-453-3. DOI: [10.1145/503376.503398](https://doi.org/10.1145/503376.503398). URL: <http://doi.acm.org/10.1145/503376.503398>.
- [GS19] Daniel Groeger and Jürgen Steimle. “LASEC: Instant Fabrication of Stretchable Circuits Using a Laser Cutter.” In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, CHI 2019, Glasgow, Scotland, UK, May 04-09, 2019*. ACM, 2019, p. 699. DOI: [10.1145/3290605.3300929](https://doi.org/10.1145/3290605.3300929). URL: <https://doi.org/10.1145/3290605.3300929>.
- [Gui87] Yves Guiard. “Asymmetric division of labor in human skilled bimanual action: the kinematic chain as a model.” In: *Journal of motor behavior* 19.4 (Dec. 1987), pp. 486–517. ISSN: 0022-2895. URL: <http://view.ncbi.nlm.nih.gov/pubmed/15136274>.
- [HC11] B. Hoddinott and J. Combs. *Drawing for dummies*. 2011.
- [HC12] Raphaël Hoarau and Stéphane Conversy. “Augmenting the Scope of Interactions with Implicit and Explicit Graphical Structures.” In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’12. Austin, Texas, USA: ACM, 2012, pp. 1937–1946. ISBN: 978-1-4503-1015-4. DOI: [10.1145/2207676.2208337](https://doi.org/10.1145/2207676.2208337). URL: <http://doi.acm.org/10.1145/2207676.2208337>.
- [HCN19] Forrest Huang, John F. Canny, and Jeffrey Nichols. “Swire: Sketch-based User Interface Retrieval.” In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. CHI ’19. Glasgow, Scotland Uk: ACM, 2019, 104:1–104:10. ISBN: 978-1-4503-5970-2. DOI: [10.1145/3290605.3300929](https://doi.org/10.1145/3290605.3300929).

- 1145/3290605.3300334. URL: <http://doi.acm.org/10.1145/3290605.3300334>.
- [Hei+10] Felix Heinrichs, Jürgen Steimle, Daniel Schreiber, and Max Mühlhäuser. "Letras: an architecture and framework for ubiquitous pen-and-paper interaction." In: *Proceedings of the 2nd ACM SIGCHI Symposium on Engineering Interactive Computing System, EICS 2010, Berlin, Germany, June 19-23, 2010*. Ed. by Noi Sukaviriya, Jean Vanderdonckt, and Michael Harrison. ACM, 2010, pp. 193–198. DOI: [10.1145/1822018.1822048](https://doi.org/10.1145/1822018.1822048). URL: <https://doi.org/10.1145/1822018.1822048>.
- [Hei+12] Felix Heinrichs, Daniel Schreiber, Jochen Huber, and Max Mühlhäuser. "Toward a Theory of Interaction in Mobile Paper-digital Ensembles." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '12. Austin, Texas, USA: ACM, 2012, pp. 1897–1900. ISBN: 978-1-4503-1015-4. DOI: [10.1145/2207676.2208328](http://doi.acm.org/10.1145/2207676.2208328). URL: <http://doi.acm.org/10.1145/2207676.2208328>.
- [Her03] Aaron Hertzmann. "Tutorial: A Survey of Stroke-Based Rendering." In: *IEEE Comput. Graph. Appl.* 23.4 (July 2003), pp. 70–81. ISSN: 0272-1716. DOI: [10.1109/MCG.2003.1210867](http://dx.doi.org/10.1109/MCG.2003.1210867). URL: <http://dx.doi.org/10.1109/MCG.2003.1210867>.
- [Her13] Oliver Herbort. "Optimal versus heuristic planning of object manipulations: A review and a computational model of the continuous end-state comfort effect." In: *New Ideas in Psychology* 31 (Dec. 2013), pp. 291–301. DOI: [10.1016/j.newideapsych.2013.01.003](http://j.newideapsych.2013.01.003).
- [Hin+05] Ken Hinckley, Patrick Baudisch, Gonzalo Ramos, and Francois Guimbretiere. "Design and Analysis of Delimiters for Selection-Action Pen Gesture Phrases in Scriboli." In: *CHI '05 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2005, pp. 451–460. ISBN: 1-58113-998-5.
- [Hin+06] Ken Hinckley, Ken Hinckley, Francois Guimbretiere, Maneesh Agrawala, Georg Apitz, and Nicholas Chen. "Phrasing Techniques for Multi-stroke Selection Gestures." In: *Proceedings of Graphics Interface 2006*. GI '06. Quebec, Canada: Canadian Information Processing Society, 2006, pp. 147–154. ISBN: 1-56881-308-2. URL: <http://dl.acm.org/citation.cfm?id=1143079.1143104>.
- [Hin+10] Ken Hinckley, Koji Yatani, Michel Pahud, Nicole Coddington, Jenny Rodenhouse, Andy Wilson, Hrvoje Benko, and Bill Buxton. "Pen + Touch = New Tools." In: *Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology*. UIST '10. New York, New York, USA: ACM, 2010, pp. 27–36. ISBN: 978-1-4503-0271-5. DOI: [10.1145/1866029.1866036](http://doi.acm.org/10.1145/1866029.1866036). URL: <http://doi.acm.org/10.1145/1866029.1866036>.
- [Hod12] Brenda Hoddinott. "Drawspace, <http://www.drawspace.com/>." In: (2012).
- [HS12] Jeffrey Heer and Ben Shneiderman. "Interactive Dynamics for Visual Analysis." In: *Communications of the ACM* 55.4 (Apr. 2012), pp. 45–54. ISSN: 0001-0782. DOI: [10.1145/2133806.2133821](http://doi.acm.org/10.1145/2133806.2133821). URL: <http://doi.acm.org/10.1145/2133806.2133821>.

- [Hur+12] Christophe Hurter, Rémi Lesbordes, Catherine Letondal, Jean-Luc Vinot, and Stéphane Conversy. "StripTIC: exploring augmented paper strips for air traffic controllers." In: *International Working Conference on Advanced Visual Interfaces, AVI 2012, Capri Island, Naples, Italy, May 22-25, 2012, Proceedings*. ACM, 2012, pp. 225–232. DOI: [10.1145/2254556.2254598](https://doi.org/10.1145/2254556.2254598). URL: <https://doi.org/10.1145/2254556.2254598>.
- [Hut+03] Hilary Hutchinson et al. "Technology Probes: Inspiring Design for and with Families." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '03. Ft. Lauderdale, Florida, USA: ACM, 2003, pp. 17–24. ISBN: 1-58113-630-7. DOI: [10.1145/642611.642616](https://doi.org/10.1145/642611.642616). URL: <http://doi.acm.org/10.1145/642611.642616>.
- [IBT13] Emmanuel Iarussi, Adrien Bousseau, and Theophanis Tsandilas. "The Drawing Assistant: Automated Drawing Guidance and Feedback from Photographs." In: *ACM Symposium on User Interface Software and Technology (UIST)*. ACM, 2013, pp. 183–192. DOI: [10.1145/2501988.2501997](https://doi.org/10.1145/2501988.2501997). URL: <http://doi.acm.org/10.1145/2501988.2501997>.
- [JDF13] Yvonne Jansen, Pierre Dragicevic, and Jean-Daniel Fekete. "Evaluating the Efficiency of Physical Visualizations." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '13. Paris, France: ACM, 2013, pp. 2593–2602. ISBN: 978-1-4503-1899-0. DOI: [10.1145/2470654.2481359](https://doi.org/10.1145/2470654.2481359). URL: <http://doi.acm.org/10.1145/2470654.2481359>.
- [Kim+09] Keun Soo Kim et al. "Large-scale pattern growth of graphene films for stretchable transparent electrodes." In: *Nature* 457:7230 (Feb. 2009), pp. 706–710. URL: <http://dx.doi.org/10.1038/nature07719>.
- [Kim+17] Nam Wook Kim, Eston Schweickart, Zhicheng Liu, Mira Dontcheva, Wilmot Li, Jovan Popovic, and Hanspeter Pfister. "Data-Driven Guides: Supporting Expressive Design for Information Graphics." In: *IEEE Transactions on Visualization and Computer Graphics* PP.99 (2017), pp. 1–1. URL: <http://dx.doi.org/10.1109/TVCG.2016.2598620>.
- [Klo+15] Clemens N. Klokmoose, James R. Eagan, Siemen Baader, Wendy Mackay, and Michel Beaudouin-Lafon. "Webstrates: Shareable Dynamic Media." In: *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. UIST '15. Charlotte, NC, USA: ACM, 2015, pp. 280–290. ISBN: 978-1-4503-3779-3. DOI: [10.1145/2807442.2807446](https://doi.org/10.1145/2807442.2807446). URL: <http://doi.acm.org/10.1145/2807442.2807446>.
- [Koh12] Matt Kohr. "Ctrl+Paint, <http://www.ctrlpaint.com/>." In: (2012).
- [KSFo6] David Kirk and Danae Stanton Fraser. "Comparing Remote Gesture Technologies for Supporting Collaborative Physical Tasks." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '06. Montrécal, Québec, Canada: ACM, 2006, pp. 1191–1200. ISBN: 1-59593-372-7. DOI: [10.1145/1124772.1124951](https://doi.org/10.1145/1124772.1124951). URL: <http://doi.acm.org/10.1145/1124772.1124951>.
- [Lan96] James A. Landay. "SILK: Sketching Interfaces Like Krazy." In: *Conference Companion on Human Factors in Computing Systems*. CHI '96. Vancouver, British Columbia, Canada: ACM, 1996, pp. 398–399. ISBN: 0-

- 89791-832-0. DOI: [10.1145/257089.257396](https://doi.org/10.1145/257089.257396). URL: <http://doi.acm.org/10.1145/257089.257396>.
- [Lia+08] Chunyuan Liao, François Guimbretière, Ken Hinckley, and Jim Hollan. "Papiercraft: A Gesture-based Command System for Interactive Paper." In: *ACM Trans. Comput.-Hum. Interact.* 14.4 (Jan. 2008), 18:1–18:27. ISSN: 1073-0516. DOI: [10.1145/1314683.1314686](https://doi.org/10.1145/1314683.1314686). URL: <http://doi.acm.org/10.1145/1314683.1314686>.
- [Lip+12] Darren J. Lipomi, Jennifer A. Lee, Michael Vosgueritchian, Benjamin C.-K. Tee, John A. Bolander, and Zhenan Bao. "Electronic Properties of Transparent Conductive Films of PEDOT:PSS on Stretchable Substrates." In: *Chemistry of Materials* 24.2 (2012), pp. 373–382. DOI: [10.1021/cm203216m](https://doi.org/10.1021/cm203216m). URL: <http://dx.doi.org/10.1021/cm203216m>.
- [Liu+18] Zhicheng Liu, John Thompson, Alan Wilson, Mira Dontcheva, James Delorey, Sam Grigg, Bernard Kerr, and John Stasko. "Data Illustrator: Augmenting Vector Design Tools with Lazy Data Binding for Expressive Visualization Authoring." In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI '18. Montreal QC, Canada: ACM, 2018, 123:1–123:13. ISBN: 978-1-4503-5620-6. DOI: [10.1145/3173574.3173697](https://doi.org/10.1145/3173574.3173697). URL: <http://doi.acm.org/10.1145/3173574.3173697>.
- [LM07] Catherine Letondal and Wendy E. Mackay. "The Paperoles Project: An analysis of paper use by music composers." In: *CoPADD, the 2nd International Workshop on Collaborating Over Paper and Digital Documents*. Proceedings of CoPADD, the 2nd International Workshop on Collaborating Over Paper and Digital Documents, London, UK. Londres, United Kingdom, 2007. URL: <https://hal.archives-ouvertes.fr/hal-01299646>.
- [Loo43] Andrew Loomis. *Figure drawing for all it's worth*. Viking Adult, 1943, p. 204.
- [LRL17] Danny Leen, Raf Ramakers, and Kris Luyten. "StrutModeling: A Low-Fidelity Construction Kit to Iteratively Model, Test, and Adapt 3D Objects." In: *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. UIST '17. Québec City, QC, Canada: ACM, 2017, pp. 471–479. ISBN: 978-1-4503-4981-9. DOI: [10.1145/3126594.3126643](https://doi.org/10.1145/3126594.3126643). URL: <http://doi.acm.org/10.1145/3126594.3126643>.
- [Lu+14] Tong Lu, Lauren Finkenauer, James Wissman, and Carmel Majidi. "Rapid Prototyping for Soft-Matter Electronics." In: *Advanced Functional Materials* 24.22 (2014), pp. 3351–3356. ISSN: 1616-3028. DOI: [10.1002/adfm.201303732](https://doi.org/10.1002/adfm.201303732). URL: <http://dx.doi.org/10.1002/adfm.201303732>.
- [LZC11] Y.J. Lee, C.L. Zitnick, and M.F. Cohen. "ShadowDraw: real-time user guidance for freehand drawing." In: *ACM Trans. on Graphics (Proc. SIGGRAPH)* 30.4 (2011), p. 27.
- [Mac+02] Wendy E. Mackay, Guillaume Pothier, Catherine Letondal, Kaare Bøegh, and Hans Erik Sørensen. "The Missing Link: Augmenting Biology Laboratory Notebooks." In: *Proceedings of the 15th Annual ACM Symposium on User Interface Software and Technology*. UIST '02. Paris, France:

- ACM, 2002, pp. 41–50. ISBN: 1-58113-488-6. DOI: [10 . 1145 / 571985 . 571992](https://doi.org/10.1145/571985.571992). URL: <http://doi.acm.org/10.1145/571985.571992>.
- [Mau+17] Nolwenn Maudet, Ghita Jalal, Philip Tchernavskij, Michel Beaudouin-Lafon, and Wendy E. Mackay. “Beyond Grids: Interactive Graphical Substrates to Structure Digital Layout.” In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. CHI ’17. Denver, Colorado, USA: ACM, 2017, pp. 5053–5064. ISBN: 978-1-4503-4655-9. DOI: [10 . 1145 / 3025453 . 3025718](https://doi.org/10.1145/3025453.3025718). URL: <http://doi.acm.org/10.1145/3025453.3025718>.
- [MH04] Tomer Moscovich and John F. Hughes. “Navigating Documents with the Virtual Scroll Ring.” In: *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology*. UIST ’04. Santa Fe, NM, USA: ACM, 2004, pp. 57–60. ISBN: 1-58113-957-8. DOI: [10 . 1145 / 1029632 . 1029642](https://doi.org/10.1145/1029632.1029642). URL: <http://doi.acm.org/10.1145/1029632.1029642>.
- [MLB12] Stefanie Mueller, Pedro Lopes, and Patrick Baudisch. “Interactive Construction: Interactive Fabrication of Functional Mechanical Devices.” In: *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*. UIST ’12. Cambridge, Massachusetts, USA: ACM, 2012, pp. 599–606. ISBN: 978-1-4503-1580-7. DOI: [10 . 1145 / 2380116 . 2380191](https://doi.org/10.1145/2380116.2380191). URL: <http://doi.acm.org/10.1145/2380116.2380191>.
- [MLG10] Sylvain Malacria, Eric Lecolinet, and Yves Guiard. “Clutch-free panning and integrated pan-zoom control on touch-sensitive surfaces: The cyclostar approach.” In: vol. 4. Jan. 2010, pp. 2615–2624. DOI: [10 . 1145 / 1753326 . 1753724](https://doi.org/10.1145/1753326.1753724).
- [MM09] David L. Morgan and Robin K. Morgan. *Single-Case Research Methods for the Behavioral and Health Sciences*. SAGE Publications, Inc., 2009. ISBN: 9781483329697. DOI: [10 . 4135 / 9781483329697](https://doi.org/10.4135/9781483329697). URL: <http://dx.doi.org/10.4135/9781483329697>.
- [Mot11] Catarina Mota. “The Rise of Personal Fabrication.” In: *Proceedings of the 8th ACM Conference on Creativity and Cognition*. C&C ’11. Atlanta, Georgia, USA: ACM, 2011, pp. 279–288. ISBN: 978-1-4503-0820-5. DOI: [10 . 1145 / 2069618 . 2069665](https://doi.org/10.1145/2069618.2069665). URL: <http://doi.acm.org/10.1145/2069618.2069665>.
- [MP94] W. Mackay and D. Pagani. “Video Mosaic: Laying out Time in a Physical Space.” In: *Proceedings of the Second ACM International Conference on Multimedia*. MULTIMEDIA ’94. San Francisco, California, USA: ACM, 1994, pp. 165–172. ISBN: 0-89791-686-7. DOI: [10 . 1145 / 192593 . 192646](https://doi.org/10.1145/192593.192646). URL: <http://doi.acm.org/10.1145/192593.192646>.
- [MSF19] Nora McDonald, Sarita Schoenebeck, and Andrea Forte. “Reliability and Inter-rater Reliability in Qualitative Research: Norms and Guidelines for CSCW and HCI Practice.” In: *Proc. ACM Hum.-Comput. Interact.* 3.CSCW (Nov. 2019), 72:1–72:23. ISSN: 2573-0142. DOI: [10 . 1145 / 3359174](https://doi.org/10.1145/3359174). URL: <http://doi.acm.org/10.1145/3359174>.

- [Nag+18] Steven Nagels, Raf Ramakers, Kris Luyten, and Wim Deferme. "Silicone Devices: A Scalable DIY Approach for Fabricating Self-Contained Multi-Layered Soft Circuits using Microfluidics." In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, CHI 2018, Montreal, QC, Canada, April 21-26, 2018*. ACM, 2018, p. 188. DOI: [10.1145/3173574.3173762](https://doi.org/10.1145/3173574.3173762). URL: <https://doi.org/10.1145/3173574.3173762>.
- [Nis+16] Gen Nishida, Ignacio Garcia-Dorado, Daniel G. Aliaga, Bedrich Benes, and Adrien Bousseau. "Interactive Sketching of Urban Procedural Models." In: *ACM Trans. Graph.* 35.4 (July 2016), 130:1–130:11. ISSN: 0730-0301. DOI: [10.1145/2897824.2925951](https://doi.org/10.1145/2897824.2925951). URL: <http://doi.acm.org/10.1145/2897824.2925951>.
- [NSI14] Makoto Nakajima, Daisuke Sakamoto, and Takeo Igarashi. "Offline Painted Media for Digital Animation Authoring." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '14. Toronto, Ontario, Canada: ACM, 2014, pp. 321–330. ISBN: 978-1-4503-2473-1. DOI: [10.1145/2556288.2557062](https://doi.org/10.1145/2556288.2557062). URL: <http://doi.acm.org/10.1145/2556288.2557062>.
- [OACo6] Sharon L. Oviatt, Alexander M. Arthur, and Julia Cohen. "Quiet interfaces that help students think." In: *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology, Montreux, Switzerland, October 15-18, 2006*. ACM, 2006, pp. 191–200. DOI: [10.1145/1166253.1166284](https://doi.org/10.1145/1166253.1166284). URL: <https://doi.org/10.1145/1166253.1166284>.
- [Oku+18] Yujiro Okuya, Nicolas Ladeveze, Cédric Fleury, and Patrick Bourdot. "ShapeGuide: Shape-Based 3D Interaction for Parameter Modification of Native CAD Data." In: *Frontiers in Robotics and AI* 5 (2018), p. 118. DOI: [10.3389/frobt.2018.00118](https://doi.org/10.3389/frobt.2018.00118). URL: <https://www.frontiersin.org/article/10.3389/frobt.2018.00118>.
- [OTA14] Halla Olafsdottir, Theophanis Tsandilas, and Caroline Appert. "Prospective Motor Control on Tabletops: Planning Grasp for Multitouch Interaction." In: *Proceedings of the 32nd International Conference on Human Factors in Computing Systems*. Toronto, Canada, Apr. 2014, pp. 2893–2902. DOI: [10.1145/2556288.2557029](https://doi.org/10.1145/2556288.2557029).
- [Ovi+12] Sharon Oviatt, Adrienne Cohen, Andrea Miller, Kumi Hodge, and Arian Mann. "The Impact of Interface Affordances on Human Ideation, Problem Solving, and Inferential Reasoning." In: *ACM Transactions on Computer-Human Interaction (TOCHI)* 19 (Oct. 2012). DOI: [10.1145/2362364.2362370](https://doi.org/10.1145/2362364.2362370).
- [OWM15] Lora Oehlberg, Wesley Willett, and Wendy E. Mackay. "Patterns of Physical Design Remixing in Online Maker Communities." In: *CHI 2015. Proceedings of ACM CHI 2015 Conference on Human Factors in Computing Systems*. ACM. Seoul, South Korea, Apr. 2015, pp. 639–648. DOI: [10.1145/2702123.2702175](https://doi.org/10.1145/2702123.2702175). URL: <https://hal.inria.fr/hal-01263171>.
- [OWS14] Simon Olberding, Michael Wessely, and Jürgen Steimle. "PrintScreen: Fabricating Highly Customizable Thin-film Touch-displays." In: *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*. UIST '14. Honolulu, Hawaii, USA: ACM, 2014, pp. 281–

290. ISBN: 978-1-4503-3069-5. DOI: [10 . 1145 / 2642918 . 2647413](https://doi.org/10.1145/2642918.2647413). URL: <http://doi.acm.org/10.1145/2642918.2647413>.
- [Pal20] Themis Palpanas. "Evolution of a Data Series Index - The iSAX Family of Data Series Indexes: iSAX, iSAX2.0, iSAX2+, ADS, ADS+, ADS-Full, ParIS, ParIS+, MESSI, DPiSAX, ULISSE, Coconut-Trie/Tree, Coconut-LSM." In: *Communications in Computer and Information Science (CCIS)* (2020).
- [Pen+18] Huaishu Peng, Jimmy Briggs, Cheng-Yao Wang, Kevin Guo, Joseph Kider, Stefanie Mueller, Patrick Baudisch, and François Guimbretière. "RoMA: Interactive Fabrication with Augmented Reality and a Robotic 3D Printer." In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI '18. Montreal QC, Canada: ACM, 2018, 579:1–579:12. ISBN: 978-1-4503-5620-6. DOI: [10.1145/3173574.3174153](https://doi.org/10.1145/3173574.3174153). URL: <http://doi.acm.org/10.1145/3173574.3174153>.
- [PG98] Terry Purcell and John S. Gero. "Drawings and the design process: A review of protocol studies in design and other disciplines and related research in cognitive psychology." In: *Design Studies* 19.4 (1998), pp. 389–430. ISSN: 0142-694X. DOI: [http://dx.doi.org/10.1016/S0142-694X\(98\)00015-5](http://dx.doi.org/10.1016/S0142-694X(98)00015-5). URL: <http://www.sciencedirect.com/science/article/pii/S0142694X98000155>.
- [Piy+16] Cecil Piya, Vinayak Vinayak, Yunbo Zhang, and Karthik Ramani. "Re-alFusion: An Interactive Workflow for Repurposing Real-World Objects towards Early-stage Creative Ideation." In: *Proceedings of Graphics Interface 2016*. GI 2016. Victoria, British Columbia, Canada: Canadian Human-Computer Communications Society / Société canadienne du dialogue humain-machine, 2016, pp. 85–92. ISBN: 978-0-9947868-1-4. DOI: [10.20380/GI2016.11](https://doi.org/10.20380/GI2016.11).
- [Qam+18] Isabel P. S. Qamar, Rainer Groh, David Holman, and Anne Roudaut. "HCI Meets Material Science: A Literature Review of Morphing Materials for the Design of Shape-Changing Interfaces." In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI '18. Montreal QC, Canada: Association for Computing Machinery, 2018, 1–23. ISBN: 9781450356206. DOI: [10.1145/3173574.3173948](https://doi.org/10.1145/3173574.3173948).
- [RBo9] Daniela Rosner and Jonathan Bean. "Learning from IKEA Hacking: I'm Not One to Decoupage a Tabletop and Call It a Day." In: *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI)*. Boston, MA, USA: ACM, 2009, pp. 419–422. ISBN: 978-1-60558-246-7. DOI: [10.1145/1518701.1518768](https://doi.org/10.1145/1518701.1518768). URL: <http://doi.acm.org/10.1145/1518701.1518768>.
- [RLB18] Donghao Ren, Bongshin Lee, and Matthew Brehmer. "Charticulator: Interactive Construction of Bespoke Chart Layouts." In: *IEEE Transactions on Visualization and Computer Graphics* PP (Aug. 2018), pp. 1–1. DOI: [10.1109/TVCG.2018.2865158](https://doi.org/10.1109/TVCG.2018.2865158).
- [Ros+12] Daniela Rosner, Jean-François Blanchette, Leah Buechley, Paul Dourish, and Melissa Mazmanian. "From Materials to Materiality: Connecting Practice and Theory in Hc." In: *CHI '12 Extended Abstracts on Human Factors in Computing Systems*. CHI EA '12. Austin, Texas, USA: ACM, 2012, pp. 2787–2790. ISBN: 978-1-4503-1016-1. DOI: [10.1145/2155701.2155702](https://doi.org/10.1145/2155701.2155702).

- 2212776 . 2212721. URL: <http://doi.acm.org/10.1145/2212776.2212721>.
- [RRS13] Bernhard Reinert, Tobias Ritschel, and Hans-Peter Seidel. "Interactive By-example Design of Artistic Packing Layouts." In: *ACM Trans. Graph.* 32.6 (Nov. 2013), 218:1–218:7. ISSN: 0730-0301. DOI: [10.1145/2508363.2508409](https://doi.org/10.1145/2508363.2508409). URL: <http://doi.acm.org/10.1145/2508363.2508409>.
- [SGHo6] Hyunyoung Song, Francois Guimbretière, and Chang Hu. "Model-Craft: capturing freehand annotations and edits on physical 3D models." In: *Proc. ACM Symposium on User Interface Software and Technology (UIST)*. ACM Press, 2006, pp. 13–22.
- [SKNo7] B. Signer, U. Kurmann, and M. Norrie. "iGesture: A General Gesture Recognition Framework." In: *Proceedings of the Ninth International Conference on Document Analysis and Recognition - Volume 02. ICDAR '07*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 954–958. ISBN: 0-7695-2822-8. URL: <http://dl.acm.org/citation.cfm?id=1304596.1304930>.
- [SM14] Allison Sauppé and Bilge Mutlu. "How Social Cues Shape Task Coordination and Communication." In: *Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work & Social Computing. CSCW '14*. Baltimore, Maryland, USA: ACM, 2014, pp. 97–108. ISBN: 978-1-4503-2540-0. DOI: [10.1145/2531602.2531610](https://doi.org/10.1145/2531602.2531610). URL: <http://doi.acm.org/10.1145/2531602.2531610>.
- [Son+06] Hyunyoung Song, François Guimbretière, Chang Hu, and Hod Lipson. "ModelCraft: Capturing Freehand Annotations and Edits on Physical 3D Models." In: *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology. UIST '06*. Montreux, Switzerland: ACM, 2006, pp. 13–22. ISBN: 1-59593-313-1. DOI: [10.1145/1166253.1166258](https://doi.org/10.1145/1166253.1166258). URL: <http://doi.acm.org/10.1145/1166253.1166258>.
- [Son+09] Hyunyoung Song, Tovi Grossman, George Fitzmaurice, François Guimbretière, Azam Khan, Ramtin Attar, and Gordon Kurtenbach. "Pen-Light: Combining a Mobile Projector and a Digital Pen for Dynamic Visual Overlay." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. CHI '09*. Boston, MA, USA: ACM, 2009, pp. 143–152. ISBN: 978-1-60558-246-7. DOI: [10.1145/1518701.1518726](https://doi.org/10.1145/1518701.1518726). URL: <http://doi.acm.org/10.1145/1518701.1518726>.
- [Son+10] Hyunyoung Song, Francois Guimbretiere, Tovi Grossman, and George Fitzmaurice. "MouseLight: Bimanual Interactions on Digital Paper Using a Pen and a Spatially-aware Mobile Projector." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. CHI '10*. Atlanta, Georgia, USA: ACM, 2010, pp. 2451–2460. ISBN: 978-1-60558-929-9. DOI: [10.1145/1753326.1753697](https://doi.org/10.1145/1753326.1753697). URL: <http://doi.acm.org/10.1145/1753326.1753697>.
- [ST97] Masaki Suwa and Barbara Tversky. "What do architects and students perceive in their design sketches? A protocol analysis." In: *Design Studies* 18.4 (1997). Descriptive models of design, pp. 385–403. ISSN: 0142-694X. DOI: [http://dx.doi.org/10.1016/S0142-694X\(97\)00008-](http://dx.doi.org/10.1016/S0142-694X(97)00008-)

2. URL: <http://www.sciencedirect.com/science/article/pii/S0142694X97000082>.
- [Tan+13] Joshua G. Tanenbaum, Amanda M. Williams, Audrey Desjardins, and Karen Tanenbaum. "Democratizing Technology: Pleasure, Utility and Expressiveness in DIY and Maker Practice." In: *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI)*. Paris, France: ACM, 2013, pp. 2603–2612. ISBN: 978-1-4503-1899-0. DOI: [10.1145/2470654.2481360](https://doi.org/10.1145/2470654.2481360). URL: <http://doi.acm.org/10.1145/2470654.2481360>.
- [TBJ15] Theophanis Tsandilas, Anastasia Bezerianos, and Thibaut Jacob. "Sketch-Sliders: Sketching Widgets for Visual Exploration on Wall Displays." In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. CHI '15. Seoul, Republic of Korea: ACM, 2015, pp. 3255–3264. ISBN: 978-1-4503-3145-6. DOI: [10.1145/2702123.2702129](https://doi.org/10.1145/2702123.2702129). URL: <http://doi.acm.org/10.1145/2702123.2702129>.
- [TD16] Theophanis Tsandilas and Pierre Dragicevic. *Accounting for Chance Agreement in Gesture Elicitation Studies*. Research Report 1584. LRI - CNRS, University Paris-Sud, Feb. 2016, p. 5. URL: <https://hal.archives-ouvertes.fr/hal-01267288>.
- [TLM09] Theophanis Tsandilas, Catherine Letondal, and Wendy Mackay. "Musink: Composing Music through Augmented Drawing." In: *International conference on Human factors in computing systems*. Boston, United States, Apr. 2009, pp. 819–828. DOI: [10.1145/1518701.1518827](https://hal.archives-ouvertes.fr/hal-00589698). URL: <https://hal.archives-ouvertes.fr/hal-00589698>.
- [TM10] Theophanis Tsandilas and Wendy E. Mackay. "Knotty Gestures: Subtle Traces to Support Interactive Use of Paper." In: *Proceedings of the International Conference on Advanced Visual Interfaces*. AVI '10. Roma, Italy: ACM, 2010, pp. 147–154. ISBN: 978-1-4503-0076-6. DOI: [10.1145/1842993.1843020](https://doi.org/10.1145/1842993.1843020). URL: <http://doi.acm.org/10.1145/1842993.1843020>.
- [TME08] Aurélien Tabard, Wendy E. Mackay, and Evelyn Eastmond. "From Individual to Collaborative: The Evolution of Prism, a Hybrid Laboratory Notebook." In: *Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work*. CSCW '08. San Diego, CA, USA: ACM, 2008, pp. 569–578. ISBN: 978-1-60558-007-4. DOI: [10.1145/1460563.1460653](https://doi.org/10.1145/1460563.1460653). URL: <http://doi.acm.org/10.1145/1460563.1460653>.
- [TMH15] Theophanis Tsandilas, Grammatikou Magdalini, and Stéphane Huot. "BricoSketch: Mixing Paper and Computer Drawing Tools in Professional Illustration." In: *International Conference on Interactive Tabletops and Surfaces (ITS'15)*. Funchal, Portugal, Nov. 2015, pp. 127–136. DOI: [10.1145/2817721.2817729](https://hal.archives-ouvertes.fr/hal-01207358). URL: <https://hal.archives-ouvertes.fr/hal-01207358>.
- [Tsa12] Theophanis Tsandilas. "Interpreting Strokes on Paper with a Mobile Assistant." In: *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*. UIST '12. Cambridge, Massachusetts, USA: ACM, 2012, pp. 299–308. ISBN: 978-1-4503-1580-7. DOI: [10.1145/2380116.2380155](https://doi.org/10.1145/2380116.2380155). URL: <http://doi.acm.org/10.1145/2380116.2380155>.

- [Tsa18] Theophanis Tsandilas. "Fallacies of Agreement: A Critical Review of Consensus Assessment Methods for Gesture Elicitation." In: *ACM Trans. Comput.-Hum. Interact.* 25.3 (June 2018), 18:1–18:49. ISSN: 1073-0516. DOI: [10.1145/3182168](https://doi.acm.org/10.1145/3182168). URL: <http://doi.acm.org/10.1145/3182168>.
- [Tsa20] Theophanis Tsandilas. "StructGraphics: Flexible Visualization Design through Data-Agnostic and Reusable Graphical Structures." In: *IEEE Transactions on Visualization and Computer Graphics*. TVCG 2021 (InfoVis 2020) (Oct. 2020). URL: <https://hal.archives-ouvertes.fr/hal-02929811>.
- [VS17] Nirzaree Vadgama and Jürgen Steimle. "Flexy: Shape-Customizable, Single-Layer, Inkjet Printable Patterns for 1D and 2D Flex Sensing." In: *Proceedings of the Eleventh International Conference on Tangible, Embedded, and Embodied Interaction*. TEI '17. Yokohama, Japan: ACM, 2017, pp. 153–162. ISBN: 978-1-4503-4676-4. DOI: [10.1145/3024969.3024989](https://doi.org/10.1145/3024969.3024989). URL: <http://doi.acm.org/10.1145/3024969.3024989>.
- [Wan+15] Jiangxin Wang, Chaoyi Yan, Kenji Jianzhi Chee, and Pooi See Lee. "Highly Stretchable and Self-Deformable Alternating Current Electroluminescent Devices." In: *Advanced Materials* 27.18 (2015), pp. 2876–2882. ISSN: 1521-4095. DOI: [10.1002/adma.201405486](https://doi.org/10.1002/adma.201405486). URL: <http://dx.doi.org/10.1002/adma.201405486>.
- [Wei+15a] Christian Weichel, John Hardy, Jason Alexander, and Hans Gellersen. "ReForm: Integrating Physical and Digital Design Through Bidirectional Fabrication." In: *Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology*. UIST '15. Daegu, Kyungpook, Republic of Korea: ACM, 2015, pp. 93–102. ISBN: 978-1-4503-3779-3. DOI: [10.1145/2807442.2807451](https://doi.org/10.1145/2807442.2807451). URL: <http://doi.acm.org/10.1145/2807442.2807451>.
- [Wei+15b] Martin Weigel, Tong Lu, Gilles Bailly, Antti Oulasvirta, Carmel Majidi, and Jürgen Steimle. "iSkin: Flexible, Stretchable and Visually Customizable On-Body Touch Sensors for Mobile Computing." In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. CHI '15. Seoul, Republic of Korea: ACM, 2015, pp. 2991–3000. ISBN: 978-1-4503-3145-6. DOI: [10.1145/2702123.2702391](https://doi.org/10.1145/2702123.2702391). URL: <http://doi.acm.org/10.1145/2702123.2702391>.
- [Wel93] Pierre Wellner. "Interacting with Paper on the DigitalDesk." In: *Commun. ACM* 36.7 (July 1993), pp. 87–96. ISSN: 0001-0782. DOI: [10.1145/159544.159630](https://doi.org/10.1145/159544.159630). URL: <http://doi.acm.org/10.1145/159544.159630>.
- [Wen10] Robert E. Wendrich. "Raw shaping form finding: Tacit tangible CAD." In: *Computer-Aided Design and Applications* 7.4 (2010), pp. 505–531. DOI: [10.3722/cadaps.2010.505-531](https://doi.org/10.3722/cadaps.2010.505-531). URL: <http://dx.doi.org/10.3722/cadaps.2010.505-531>.
- [Wer11] M. Werner. *Model Making*. Architecture briefs. Princeton Architectural Press, 2011. ISBN: 9781616893194. URL: <https://books.google.fr/books?id=q0lTnQAACAAJ>.

- [Wes18] Michael Wessely. “Fabricating Malleable Interaction-Aware Materials.” Theses. Université Paris-Saclay, Dec. 2018. URL: <https://tel.archives-ouvertes.fr/tel-01988897>.
- [WHA07] Wesley Willett, Jeffrey Heer, and Maneesh Agrawala. “Scented Widgets: Improving Navigation Cues with Embedded Visualizations.” In: *IEEE Transactions on Visualization and Computer Graphics* 13.6 (Nov. 2007), pp. 1129–1136. ISSN: 1077-2626. DOI: [10.1109/TVCG.2007.70589](https://doi.org/10.1109/TVCG.2007.70589). URL: <http://dx.doi.org/10.1109/TVCG.2007.70589>.
- [Wil+11] Karl D.D. Willis, Cheng Xu, Kuan-Ju Wu, Golan Levin, and Mark D. Gross. “Interactive Fabrication: New Interfaces for Digital Fabrication.” In: *Proceedings of the Fifth International Conference on Tangible, Embedded, and Embodied Interaction*. TEI ’11. Funchal, Portugal: ACM, 2011, pp. 69–72. ISBN: 978-1-4503-0478-8. DOI: [10.1145/1935701.1935716](https://doi.org/10.1145/1935701.1935716). URL: <http://doi.acm.org/10.1145/1935701.1935716>.
- [Wino6] Megan Winget. “Heroic Frogs Save the Bow: Performing Musician’s Annotation and Interaction Behavior with Written Music.” In: *Proceedings of the 7th International Conference on Music Information Retrieval*. Victoria (BC), Canada, 2006.
- [Wrio5] Matthew Wright. “Open Sound Control: An Enabling Technology for Musical Networking.” In: *Org. Sound* 10.3 (Dec. 2005), pp. 193–200. ISSN: 1355-7718. DOI: [10.1017/S1355771805000932](https://doi.org/10.1017/S1355771805000932). URL: <http://dx.doi.org/10.1017/S1355771805000932>.
- [WTM16] Michael Wessely, Theophanis Tsandilas, and Wendy E. Mackay. “Stretchis: Fabricating Highly Stretchable User Interfaces.” In: *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. UIST ’16. Tokyo, Japan: ACM, 2016, pp. 697–704. ISBN: 978-1-4503-4189-9. DOI: [10.1145/2984511.2984521](https://doi.org/10.1145/2984511.2984521). URL: <http://doi.acm.org/10.1145/2984511.2984521>.
- [WTM18] Michael Wessely, Theophanis Tsandilas, and Wendy Mackay. “Shape-Aware Material: Interactive Fabrication with ShapeMe.” In: *UIST 2018 - 31st ACM Symposium on User Interface Software and Technology*. Vol. 18. Berlin, Germany: ACM Press, Oct. 2018, pp. 127–139. DOI: [10.1145/3242587.3242619](https://hal.archives-ouvertes.fr/hal-01872776). URL: <https://hal.archives-ouvertes.fr/hal-01872776>.
- [Xia+18a] Haijun Xia, Nathalie Henry Riche, Fanny Chevalier, Bruno De Araujo, and Daniel Wigdor. “DataInk: Direct and Creative Data-Oriented Drawing.” In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI ’18. Montreal QC, Canada: ACM, 2018, 223:1–223:13. ISBN: 978-1-4503-5620-6. DOI: [10.1145/3173574.3173797](https://doi.org/10.1145/3173574.3173797). URL: <http://doi.acm.org/10.1145/3173574.3173797>.
- [Xia+18b] Haijun Xia, Sebastian Herscher, Ken Perlin, and Daniel Wigdor. “Space-time: Enabling Fluid Individual and Collaborative Editing in Virtual Reality.” In: *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*. UIST ’18. Berlin, Germany: ACM, 2018, pp. 853–866. ISBN: 978-1-4503-5948-1. DOI: [10.1145/3242587.3242597](https://doi.org/10.1145/3242587.3242597). URL: <http://doi.acm.org/10.1145/3242587.3242597>.

- [Xie+14] Jun Xie, Aaron Hertzmann, Wilmot Li, and Holger Winnemöller. "PortraitSketch: Face Sketching Assistance for Novices." In: *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*. UIST '14. Honolulu, Hawaii, USA: ACM, 2014, pp. 407–417. ISBN: 978-1-4503-3069-5. DOI: [10.1145/2642918.2647399](https://doi.org/10.1145/2642918.2647399). URL: <http://doi.acm.org/10.1145/2642918.2647399>.
- [Yan+14] Chaoyi Yan, Wenbin Kang, Jiangxin Wang, Mengqi Cui, Xu Wang, Ce Yao Foo, Kenji Jianzhi Chee, and Pooi See Lee. "Stretchable and wearable electrochromic devices." In: *ACS Nano* 8.1 (2014), pp. 316 – 322. ISSN: 1936-086X.
- [YPRKo8] Ron Yeh, Andreas Paepcke, and Scott R. Klemmer. "Iterative design and evaluation of an event architecture for pen-and-paper interfaces." In: Jan. 2008, pp. 111–120. DOI: [10.1145/1449715.1449734](https://doi.org/10.1145/1449715.1449734).
- [Yue+17] Ya-Ting Yue, Xiaolong Zhang, Yongliang Yang, Gang Ren, Yi-King Choi, and Wenping Wang. "WireDraw: 3D Wire Sculpturing Guided with Mixed Reality." In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. CHI '17. Denver, Colorado, USA: ACM, 2017, pp. 3693–3704. ISBN: 978-1-4503-4655-9. DOI: [10.1145/3025453.3025792](https://doi.org/10.1145/3025453.3025792). URL: <http://doi.acm.org/10.1145/3025453.3025792>.
- [Zha+07] Shengdong Zhao, Pierre Dragicevic, Mark Chignell, Ravin Balakrishnan, and Patrick Baudisch. "Earpod: eyes-free menu selection using touch input and reactive audio feedback." In: *Proceedings of the Conference on Human Factors in Computing Systems (CHI '07)*. ACM Press, 2007, pp. 1395–1404.
- [ZMo6] Robert C. Zeleznik and Timothy S. Miller. "Fluid inking: augmenting the medium of free-form inking with gestures." In: *Graphics Interface*. 2006.
- [ZP13] Amit Zoran and Joseph A. Paradiso. "FreeD: A Freehand Digital Sculpting Tool." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '13. Paris, France: ACM, 2013, pp. 2613–2616. ISBN: 978-1-4503-1899-0. DOI: [10.1145/2470654.2481361](https://doi.org/10.1145/2470654.2481361). URL: <http://doi.acm.org/10.1145/2470654.2481361>.